

TATIANA RENATA GARCIA

**MODELAGEM E CONTROLE DA COORDENAÇÃO
DO TRÁFEGO URBANO ATRAVÉS DE
FORMALISMOS PARA SISTEMAS A EVENTOS
DISCRETOS E HÍBRIDOS**

FLORIANÓPOLIS

2007

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

**MODELAGEM E CONTROLE DA COORDENAÇÃO
DO TRÁFEGO URBANO ATRAVÉS DE
FORMALISMOS PARA SISTEMAS A EVENTOS
DISCRETOS E HÍBRIDOS**

Tese submetida à
Universidade Federal de Santa Catarina
como parte dos requisitos para a
obtenção do grau de Doutor em Engenharia Elétrica.

TATIANA RENATA GARCIA

Florianópolis, agosto de 2007.

MODELAGEM E CONTROLE DA COORDENAÇÃO DO TRÁFEGO URBANO ATRAVÉS DE FORMALISMOS PARA SISTEMAS A EVENTOS DISCRETOS E HÍBRIDOS

Tatiana Renata Garcia

Esta Tese foi julgada adequada para a obtenção do título de Doutor em Engenharia Elétrica, Área de Concentração em *Automação e Sistemas*, e aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina.

Prof. José Eduardo Ribeiro Cury, Dr.
Orientador

Prof. Werner Kraus Junior, Dr.
Co-Orientador

Profa. Katia Campos de Almeida, Dra.
Coordenadora do Programa de Pós-Graduação em Engenharia Elétrica

Banca Examinadora:

Prof. José Eduardo Ribeiro Cury, Dr.
Presidente

Prof. Werner Kraus Junior, Dr.

Prof. Rafael Santos Mendes, Dr.

Ricardo David Katz, Dr.

Prof. Jean-Marie Farines, Dr.

Prof. Eduardo Camponogara, Dr.

Ao meu marido Rafael e à minha querida família.

AGRADECIMENTOS

Primeiramente agradeço a Deus, aquele que sempre me acompanha.

Gostaria de agradecer ao Rafael, que no início desta aventura era meu noivo e agora é meu marido, ele certamente teve um papel fundamental para a conclusão desta tese. Assim como ele, minha mãe e minha irmã foram pessoas muito importantes em todos os momentos.

Sinceros agradecimentos ao meu orientador José Cury, pessoa admirável tanto no âmbito acadêmico como pessoal. Além dele, meu segundo orientador, Werner Kraus Junior, contribuiu muito para o término bem sucedido deste trabalho. Eles foram acima de tudo amigos ao longo de todos estes anos.

Não posso deixar de agradecer aos colegas do DAS, em especial a amizade da Luciana Bolan Frigo, Cássia Yuri Tatibana, Michelle Wangham, Patrícia Pena, Cristiane Paim, Karina Barbosa e André Leal, com certeza amigos para toda vida. Além deles, outros grandes amigos foram importantes ao longo destes anos: Luiz Magno, Cleicio Martins, Edson de Pieri, Alysson Neves Bessani, José Eduardo Brandão, Jerusa Marchi, Paulo Mafra, Fábio Favarim, Fábio Rocha, Emerson de Mello, Ricardo Schmidt, Rodrigo Sumar, Marcos Vallim, Ana Barbara, além daqueles que porventura esqueci. Agradeço aos amigos que fiz na minha estada em Toulouse na França: Magnos Martinello, Roberta Gomes, José Valentim, Alessandra Cristina, Eduardo, Francisco e Cristina.

Agradeço aos colegas do grupo de SEDs, pessoas que ajudaram a enriquecer este trabalho, entre eles o Antonio Carrilho, Max de Queiroz, César Torrico, Christianne Reiser, Rodrigo Braga, Frederico Mello e Gustavo Bouzon.

Aos professores que participaram da banca, pelas suas valiosas críticas e sugestões. Agradeço também aos professores e funcionários do DAS, assim como aos funcionários da secretaria do PPGEEL Wilson e Marcelo por toda a ajuda prestada ao longo destes anos.

À CAPES pelo apoio financeiro no Brasil e ao convênio CAPES/COFECUB pelo estágio no LAAS, em Toulouse na França.

Resumo da Tese apresentada à UFSC como parte dos requisitos necessários para obtenção do grau de Doutor em Engenharia Elétrica.

MODELAGEM E CONTROLE DA COORDENAÇÃO DO TRÁFEGO URBANO ATRAVÉS DE FORMALISMOS PARA SISTEMAS A EVENTOS DISCRETOS E HÍBRIDOS

Tatiana Renata Garcia

Agosto/2007

Orientador: Prof. José Eduardo Ribeiro Cury, Dr.

Co-orientador: Prof. Werner Kraus Junior, Dr.

Área de Concentração: Automação e Sistemas

Palavras-chave: coordenação semafórica, controle de tráfego urbano, autômatos híbridos, verificação formal, álgebra max-plus, (A,B)-invariância

Número de Páginas: xiv + 148

A coordenação semafórica é um problema importante para o funcionamento dos sistemas de tráfego urbano. Nesta tese, examina-se o uso de métodos de modelagem e controle de sistemas a eventos discretos (SED) e sistemas híbridos (SH) para resolver este problema. Primeiramente o problema de coordenação é modelado utilizando autômatos híbridos. O objetivo é a análise do comportamento do sistema através de resultados de simulação e verificação formal de propriedades. Em um segundo momento, os sistemas de tráfego são modelados como sistemas puramente discretos. A álgebra max-plus é utilizada para modelar e controlar vias arteriais. Através da abordagem discreta é possível além de analisar o comportamento do sistema realizar síntese de controle. A síntese é realizada utilizando os conceitos de (A,B)-invariância, e os resultados obtidos são aplicados em um modelo de tráfego para simulação. Os resultados obtidos através da abordagem híbrida e da abordagem discreta demonstram que é possível utilizar os métodos propostos para SEDs e SHs para resolver o problema de coordenação semafórica. Os métodos obtidos podem vir a compor ferramentas para a análise e tomada de decisão para apoiar engenheiros de tráfego.

Abstract of Thesis presented to UFSC as a partial fulfillment of the requirements for the degree of Doctor in Electrical Engineering.

MODELING AND CONTROL OF URBAN TRAFFIC COORDINATION USING FORMALISMS FOR HYBRID AND DISCRETE-EVENT SYSTEMS

Tatiana Renata Garcia

August/2007

Advisor: Prof. José Eduardo Ribeiro Cury, Dr.

Co-advisor: Prof. Werner Kraus Junior, Dr.

Area of Concentration: Automation and Systems

Key words: traffic light coordination, urban traffic control, hybrid automaton, formal verification, max-plus algebra, (A,B)-invariance

Number of Pages: xiv + 148

Traffic light coordination is a key aspect in the operation of urban traffic systems. In this thesis we apply formalisms for the modeling and control of discrete event systems (DES) and hybrid systems (HS) to the problem of traffic light coordination. First, the problem is modeled as a hybrid system using hybrid automata. The goal is to analyze the system behavior through simulation and the formal verification of properties. We then model traffic systems using a pure discrete approach. Arterial networks are modelled and controlled using max-plus algebra. In addition to the analysis of the system behavior, this approach allows us to realize controller synthesis. This synthesis uses (A,B)-invariance concepts, and its results are applied in a traffic simulation model. The results obtained using the hybrid and discrete approaches show that it is possible to use the formalisms developed for DES and HS to solve the problem of traffic light coordination. These models can be used by traffic engineers during traffic systems analysis to provide key insights on the various possible solutions.

Sumário

1	Introdução	1
1.1	Motivação	1
1.2	Objetivos	3
1.3	Resultados Obtidos	3
1.4	Organização do Texto	4
2	Coordenação Semafórica	6
2.1	Coordenação entre Interseções	6
2.2	Técnicas de Coordenação	7
2.3	Modelos de Filas	10
2.3.1	Modelo de Fila Vertical	10
2.3.2	Modelo de Fila Horizontal	12
2.3.3	Comparação Entre os Modelos de Fila Vertical e Horizontal	13
2.4	A Coordenação em Sistemas Integrados de Controle de Tráfego	13
2.5	Conclusão	16
3	Sistemas Híbridos	17
3.1	Exemplo de um Sistema Híbrido	17
3.2	Estudos em Sistemas Híbridos	19
3.3	Autômatos Híbridos	21
3.3.1	Definição Formal	21
3.3.2	Alcançabilidade em Autômatos Híbridos	24

3.4	Verificação de Sistemas Híbridos	27
3.4.1	Métodos de Verificação Formal	27
3.4.2	Abordagens Algorítmicas de Verificação	28
3.5	Ferramentas	29
3.6	Conclusão	31
4	Modelagem e Análise de Sistemas de Tráfego como Sistemas Híbridos	32
4.1	Trabalhos Relacionados	32
4.2	Modelagem em Alto Nível	35
4.2.1	Sistemas Compostos por Interseções Conectadas	39
4.2.2	Aplicação da Metodologia a uma Interseção Isolada	40
4.3	Modelos Refinados	42
4.3.1	Exemplo 1 – Sistema de uma Interseção Isolada	43
4.3.2	Exemplo 2 – Sistema com Duas Interseções Conectadas	44
4.4	Implementação	50
4.4.1	Ferramenta Utilizada – <i>CheckMate</i>	50
4.4.2	Simulação – Resultados Obtidos	52
4.5	Verificação de Propriedades	61
4.5.1	Verificação 1 – Bloqueio	61
4.5.2	Verificação 2 – Tempo Ocioso de Verde	64
4.6	Conclusão	66
5	Álgebra Max-Plus	68
5.1	Exemplo de um Sistema Dinâmico Max-Plus Linear	68
5.2	Teoria de Dióides	72
5.2.1	Modelagem de GET através de Dióides	75
5.2.2	Teoria da Residuação	76
5.2.3	Teoria Spectral de Matrizes no \mathbb{R}_{\max}	77

5.3	Semimódulos (A,B)-Invariantes	78
5.3.1	Equação $Ax = Bx$ sobre o \mathbb{R}_{\max}	80
5.3.2	Semimódulos (geometricamente) (A,B)-Invariantes	80
5.3.3	Semimódulos (algebricamente) (A,B)-Invariantes	83
5.3.4	Exemplo	84
5.4	Conclusão	86
6	Modelagem e Controle de Sistemas de Tráfego através da Álgebra Max-Plus	87
6.1	Trabalhos Relacionados	87
6.2	Proposta para Modelagem e Controle de Sistemas de Tráfego	93
6.2.1	Modelo Genérico para n Interseções	99
6.2.2	Exemplo	101
6.3	Implementação do Controle	106
6.3.1	Obtenção dos Vetores $x(k)$	106
6.3.2	Robustez do Controlador	108
6.4	Experimentos	110
6.4.1	Modelo de Tráfego para Simulação	110
6.4.2	Algoritmo para Simulação	112
6.4.3	Simulação 1	115
6.4.4	Simulação 2	120
6.4.5	Simulação 3	121
6.4.6	Simulação 4	123
6.5	Conclusão	128
7	Conclusões	130
7.1	Revisão dos Objetivos	130
7.2	Contribuições e Resultados da Tese	130
7.3	Perspectivas Futuras	133
A	Terminologia de Tráfego Urbano	135

Lista de Figuras

2.1	Situação desejada para coordenação entre interseções	6
2.2	Diagrama espaço-tempo	8
2.3	Formação e descarga de filas	11
2.4	Chegadas e partidas de veículos acumulados para um caso ideal	12
2.5	Modelo de fila vertical	13
2.6	Fila horizontal	14
2.7	Comparação entre fila vertical e horizontal	15
3.1	Modelo para o termostato	18
3.2	Trajetória para a temperatura do quarto	18
3.3	Representação gráfica de um autômato híbrido	22
3.4	Autômato híbrido de três estados e seu comportamento	23
3.5	Sucessores discretos de F	25
3.6	Estados alcançáveis depois de um tempo r	25
3.7	Operadores predecessores contínuos e discretos	26
4.1	Interseção isolada	34
4.2	Tipos de <i>links</i> que se conectam a um nó	36
4.3	Exemplo de uma interseção isolada	37
4.4	Diagrama de estágios e fases para interseção isolada	38
4.5	Exemplo de conexão em uma interseção isolada	40
4.6	Exemplo 1 – interseção isolada	41

4.7	Estágios e fases	41
4.8	Autômato em alto nível para uma interseção isolada	42
4.9	Autômato do modelo refinado para uma interseção isolada	44
4.10	Autômato final para o modelo refinado (uma interseção isolada)	45
4.11	Sistema com duas interseções conectadas	45
4.12	<i>Links</i> para o sistema com duas interseções	45
4.13	Modelos em alto nível para S1 e S2	46
4.14	Estágios e fases – exemplo 2	46
4.15	Relógios para modelar guardas	48
4.16	Autômato refinado para segunda interseção	49
4.17	Funcionamento das guardas considerando o tempo de deslocamento	49
4.18	Autômato do semáforo S1 considerando as dinâmicas dos relógios	50
4.19	Blocos disponíveis no <i>CheckMate</i>	51
4.20	Modelo desenvolvido no <i>CheckMate</i>	54
4.21	FSMB que modela semáforo S1	55
4.22	FSMB que modela semáforo S2	55
4.23	Comportamento das filas no semáforo S1 – Simulação 1	56
4.24	Comportamento das filas no semáforo S2 – Simulação 1	57
4.25	Evolução dos estados nos FSMBs – Simulação 1	57
4.26	Comportamento dos relógios – Simulação 1	58
4.27	Comportamento das filas no semáforo S1 – Simulação 2	59
4.28	Comportamento das filas no semáforo S2 – Simulação 2	59
4.29	Evolução dos estados nos FSMBs – Simulação 2	60
4.30	Comportamento dos relógios – Simulação 2	60
4.31	FSMB que modela semáforo para interseção isolada	62
4.32	Comportamento das filas – Verificação 1	63
4.33	Comportamento das filas – Verificação 1 com fluxos alterados	64

4.34	FSMB modificada que modela semáforo para interseção isolada	65
4.35	Comportamento das filas – Verificação 2	66
5.1	Rede de trens entre duas cidades	69
5.2	Grafo para rede de trens entre duas cidades	70
5.3	GET para a rede de trens entre duas cidades	72
5.4	Configurações para transições de GET	75
6.1	Estágios para um cruzamento normal e um T-cruzamento	88
6.2	Pequeno exemplo	88
6.3	GET para pequeno exemplo	89
6.4	Exemplo – rede de trens	90
6.5	Sistema mínimo (2 interseções)	93
6.6	Modelo de uma via de mão dupla	94
6.7	Modelo de um semáforo	94
6.8	Conexão de uma via e um semáforo	95
6.9	Modelo de uma malha semaforizada	95
6.10	Modelo de uma malha com entradas de controle	95
6.11	Grafo de precedência para matriz E da especificação	98
6.12	Malha viária controlada	99
6.13	Malha viária com n interseções	99
6.14	grafo de eventos para arterial com n interseções	100
6.15	Malha viária com 3 interseções	101
6.16	Modelo com entradas de controle	101
6.17	Aproximações ou <i>links</i> em via de mão dupla	111
6.18	Seções de uma faixa	111
6.19	Exemplo de três interseções e diagrama de <i>links</i>	112
6.20	Evolução das filas $x_1(t)$ em função das chegadas $a_{11}(t)$ e da sinalização $m_1(t)$	116

6.21	Evolução das filas $x_2(t)$ em função das chegadas $a_{21}(t)$ e da sinalização $m_2(t)$	116
6.22	Evolução das filas $x_3(t)$ em função das chegadas $a_{31}(t)$ e da sinalização $m_3(t)$	117
6.23	Evolução das filas $x_4(t)$ em função das chegadas $a_{41}(t)$ e da sinalização $m_4(t)$	117
6.24	Evolução das filas $x_5(t)$ em função das chegadas $a_{51}(t)$ e da sinalização $m_5(t)$	118
6.25	Evolução das filas $x_6(t)$ em função das chegadas $a_{61}(t)$ e da sinalização $m_6(t)$	118
6.26	Banda de verde gerada – Simulação 1	119
6.27	Nova evolução das filas $x_1(t)$ em função das chegadas $a_{11}(t)$ e da sinalização $m_1(t)$.	124
6.28	Nova evolução das filas $x_2(t)$ em função das chegadas $a_{21}(t)$ e da sinalização $m_2(t)$.	124
6.29	Nova evolução das filas $x_3(t)$ em função das chegadas $a_{31}(t)$ e da sinalização $m_3(t)$.	125
6.30	Nova evolução das filas $x_4(t)$ em função das chegadas $a_{41}(t)$ e da sinalização $m_4(t)$.	125
6.31	Nova evolução das filas $x_5(t)$ em função das chegadas $a_{51}(t)$ e da sinalização $m_5(t)$.	126
6.32	Nova evolução das filas $x_6(t)$ em função das chegadas $a_{61}(t)$ e da sinalização $m_6(t)$.	126
6.33	Banda de verde gerada – Simulação 4	126
A.1	Diagrama de estágios – exemplo	136
A.2	Conjunto de movimentos	137
A.3	Definição de via, pista e faixa	138

Lista de Tabelas

4.1	Matriz de incidência A	37
4.2	Matriz de incidência do semáforo S	41
4.3	Guardas do autômato para uma interseção isolada	44
4.4	Matriz de incidência do semáforo $S1$	45
4.5	Matriz de incidência do semáforo $S2$	46
4.6	Guardas do autômato do semáforo $S2$	48
4.7	Guardas do autômato para uma interseção isolada	61
6.1	Matriz W para especificação	98
6.2	Matriz $E = (-W)^T$	98
6.3	Matriz A para exemplo de 3 interseções	102
6.4	Matriz B para exemplo de 3 interseções	102
6.5	Matriz W para exemplo de três interseções	103
6.6	Matriz E para exemplo de três interseções	104
6.7	Matriz com conjunto de estados iniciais K^\uparrow	105
6.8	Matriz de controle F_1	105
6.9	Matriz de controle F_2	106
6.10	Resultados encontrados para as restrições – Simulação 1	120
6.11	Resultados encontrados para as restrições – Simulação 2	121
6.12	Resultados encontrados para as restrições – Simulação 3	122
6.13	Matriz com conjunto de estados iniciais K'	123
6.14	Resultados encontrados para as restrições – Simulação 4	127

Capítulo 1

Introdução

1.1 Motivação

Pela importância que tem na vida das pessoas, os sistemas de tráfego veicular urbano necessitam de técnicas de controle que garantam seu melhor desempenho. Em particular, um aspecto importante do funcionamento do sistema de tráfego urbano é a coordenação das indicações semafóricas entre interseções adjacentes, garantindo progressão veicular sempre que possível. A progressão veicular é desejada pois permite reduzir o número de paradas dos motoristas durante o seu deslocamento, aumentando o seu conforto. Além disso, possibilita a redução dos ruídos, dos custos com combustíveis e da poluição gerada pela aceleração e desaceleração dos veículos.

Ao observar-se o deslocamento de veículos pelas vias de uma cidade é possível perceber que ali atua um sistema híbrido, no qual coexistem dinâmicas contínuas e dinâmicas discretas. As dinâmicas contínuas são identificadas, por exemplo, no crescimento e diminuição do número de veículos ao longo das vias, e nos deslocamentos veiculares. Já dinâmicas discretas aparecem, por exemplo, nas trocas semafóricas, identificação de bloqueios nas vias e chegadas de pelotões de veículos em um cruzamento. Dependendo da abordagem algumas dinâmicas podem ser definidas tanto como discretas como contínuas.

A partir desta constatação, surge, naturalmente, a indagação de quais resultados da Teoria de Sistemas Híbridos e da Teoria de Sistemas a Eventos Discretos poderiam ser aplicados na descrição e operação dos sistemas de tráfego urbano. Outra pergunta que pode ser feita é se os formalismos utilizados para modelar e controlar sistemas híbridos (SHs) e sistemas a eventos discretos (SEDs) podem vir a ser mais adequados do que os disponíveis para tratar o problema de modelagem e controle de sistemas de tráfego urbano.

Atualmente se encontram poucas respostas na literatura de controle de tráfego urbano para as questões apontadas, e isto ocorre devido a dois motivos:

- os formalismos e métodos de controle propostos para sistemas híbridos e a eventos discretos

são relativamente recentes, tendo sido desenvolvidos com mais intensidade a partir dos anos 1980 encontrando aplicação principalmente em problemas de automação industrial; e

- os trabalhos sobre modelagem e controle de tráfego urbano tem origem em métodos de otimização e técnicas clássicas de modelagem, como as equações de escoamento de fluidos e os modelos de seguimento veicular baseados em equações diferenciais [Gerlough e Huber, 1975; Gazis, 2002].

As questões apresentadas motivam a busca de soluções para o problema da coordenação dos sistemas de tráfego urbano através das técnicas de análise e síntese de controle de sistemas híbridos e de sistemas a eventos discretos. Em particular, nesta tese, dois aspectos do problema são considerados:

- a análise do comportamento das filas geradas no sistema, visando avaliar a coordenação semafórica e verificar propriedades;
- a síntese de controle para garantir progressão veicular ao longo de vias arteriais.

Existem diversas técnicas de coordenação na literatura que são baseadas no comportamento das filas. Os formalismos para sistemas híbridos permitem modelar os aspectos contínuos dos sistemas, portanto é satisfatório utilizá-los quando se deseja modelar as filas. Além disso, os formalismos permitem realizar verificação formal de propriedades e então a capacidade de análise do comportamento do sistema é incrementada. A verificação de propriedades é interessante pois fornece ao engenheiro de tráfego informações que garantem que determinadas situações nunca acontecerão, diminuindo assim a complexidade do projeto de controle semafórico. Existem na literatura alguns trabalhos que modelam os sistemas de tráfego urbano como sistemas híbridos, entre eles, os trabalhos de Lei e Ozguner [2001], Febbraro e Sacco [2004]; Febbraro et al. [2001, 2003] e Tolba et al. [2001], e em nenhum deles o problema de verificação formal de propriedades e síntese de controle é tratado. Os trabalhos citados apresentam propostas de modelagem para os sistemas de tráfego e os modelos são avaliados através de resultados de simulação. Para simular os modelos são utilizadas estratégias de controle baseadas em métodos de otimização.

Uma outra abordagem do problema de coordenação não diz respeito ao comportamento das filas. Estas técnicas preocupam-se em sincronizar os semáforos de maneira que pelotões de veículos possam encontrar a maioria dos semáforos em verde, garantido assim a progressão veicular. Um pelotão pode ser interpretado como um ponto, ou uma entidade indivisível, e então imagina-se que modelos puramente discretos podem ser promissores para resolver o problema de síntese de controle. Um sistema a eventos discretos é um sistema dinâmico que evolui de acordo com a ocorrência abrupta de eventos físicos, em intervalos de tempo em geral irregulares e desconhecidos [Cassandras e Lafortune, 1999]. Existem diversos formalismos que permitem modelar os SEDs, dentre eles é possível citar as redes de Petri, cadeias de Markov, teoria de linguagens e autômatos, teoria de dióides, e outros.

Um formalismo que tem sido aplicado com sucesso na modelagem de sistemas sujeitos a restrições de sincronização é a álgebra max-plus [Baccelli et al., 1992], onde os sistemas são modelados

como sistemas dinâmicos max-plus lineares. Os sistemas de tráfego são sistemas onde a sincronização é necessária, portanto a álgebra max-plus pode ser utilizada com sucesso para representá-los.

Muitos resultados básicos da teoria de controle clássico tem sua versão análoga em álgebra max-plus. Em Katz [2007, 2003], por exemplo, a abordagem geométrica proposta por Wonham [1985] foi adaptada para os semi-anéis max-plus. Assim como a formulação e solução de muitos problemas é realizada através de espaços invariantes na teoria clássica, Katz [2007] formula o problema de controle de SEDs através de semimódulos invariantes.

Com base nas afirmações anteriores, define-se nesta tese o problema de síntese de controle com coordenação semafórica como um problema de controle de SEDs. Tal problema tem sido tratado por técnicas baseadas em métodos de otimização, por exemplo, o MULTIBAND [Gartner et al., 1990]. O uso de formalismos de SEDs pode permitir a obtenção de modelos que descrevam explicitamente as especificações da lógica de controle desejada pelo engenheiro de tráfego, algo nem sempre possível em outros métodos.

1.2 Objetivos

O problema abordado nesta tese é a coordenação das indicações semafóricas entre interseções, onde se deseja estudar e examinar o uso de técnicas de sistemas a eventos discretos e híbridos para modelar e controlar os sistemas de tráfego urbano.

Esta tese possui dois objetivos. O primeiro objetivo é propor uma técnica de modelagem para os sistemas de tráfego urbano baseada na teoria de sistemas híbridos, onde os modelos permitem realizar análise do sistema por simulação e verificação de propriedades. O formalismo adotado para o desenvolvimento desta técnica é o autômato híbrido.

O segundo objetivo desta tese é resolver o problema de modelagem e síntese de controle com sincronização semafórica dos sistemas de tráfego urbano através da álgebra max-plus. Especificamente, o problema estudado é o da coordenação semafórica de vias arteriais. Vias arteriais são as principais vias de uma cidade, onde a fluidez do tráfego é desejada pois o fluxo veicular é intenso e rápido. A solução proposta utiliza o conceito de (A,B)-invariância para obter um sistema controlado com realimentação de estados.

1.3 Resultados Obtidos

Ao longo desta tese a coordenação semafórica dos sistemas de tráfego urbano é estudada sob a ótica dos sistemas híbridos e sob a ótica dos sistemas a eventos discretos.

Os resultados obtidos ao modelar os sistemas de tráfego como sistemas híbridos são:

- a proposta de uma metodologia de modelagem dos sistemas de tráfego utilizando os autômatos híbridos. Cada estado do autômato híbrido representa um estágio do semáforo (conceitos sobre

tráfego em Apêndice A), sendo desenvolvidos modelos individuais para cada cruzamento semaforizado do sistema. O acoplamento do sistema é realizado através da definição de guardas do autômato híbrido, permitindo assim a coordenação semafórica;

- além de obter modelos para simulação do sistema, é possível realizar verificação formal de algumas propriedades, ou seja, realizar uma análise do sistema. O número de interseções do sistema influencia diretamente na possibilidade de realizar verificação formal de propriedades, devido à complexidade computacional envolvida no cálculo do espaço de estados. As propriedades verificadas nesta tese são traduzidas em um problema de alcançabilidade nos autômatos desenvolvidos.

Ao estudar o sistema de tráfego como um sistema a eventos discretos as variáveis envolvidas na modelagem são diferentes pois apenas dinâmicas discretas são consideradas. O principal objetivo é implementar um controle com realimentação de estados. Os principais resultados encontrados são:

- a proposta de uma metodologia de modelagem dos sistemas de tráfego utilizando a álgebra max-plus. O sistema é modelado através de grafos a eventos temporizados, e o problema de coordenação semafórica é definido através de um conjunto de restrições que são definidas sobre as transições do grafo de eventos. O método proposto por Katz [2007] é utilizado para obter uma matriz de realimentação de estados que implementa o controle;
- aspectos relacionados à robustez do controle ao implementá-lo em um sistema real são discutidos. A fim de avaliar os controladores obtidos os modelos max-plus lineares são implementados em um modelo de tráfego para simulação.

Estes resultados foram apresentados à comunidade científica através das seguintes publicações: Garcia et al. [2007, 2006a,b]; Garcia e Cury [2004b,a]. Além disso, os resultados de simulação obtidos no Capítulo 6 serão submetidos para uma revista.

1.4 Organização do Texto

Este capítulo inicial apresentou a motivação do trabalho, os seus objetivos e um resumo dos principais resultados. A organização do restante deste documento é apresentada a seguir.

No Capítulo 2 discute-se a coordenação semafórica, problema abordado ao longo da tese. São apresentados técnicas de coordenação, modelos de filas e sistemas de controle existentes na literatura.

O Capítulo 3 apresenta os sistemas híbridos. O formalismo adotado neste trabalho é descrito, no caso os autômatos híbridos. Exemplos de sistemas híbridos e um breve histórico da pesquisa nesta área também são citados.

Os resultados obtidos ao estudar os sistemas de tráfego como sistemas híbridos estão no Capítulo 4. Uma proposta de modelagem para os sistemas através dos autômatos híbridos, bem como resultados

de simulação e verificação obtidos através da ferramenta *CheckMate* também são apresentados neste capítulo.

No Capítulo 5 a álgebra max-plus é estudada. São apresentados os conceitos básicos relacionados com o formalismo. Além disso, conceitos relacionados aos semimódulos (A,B)-invariantes também são descritos.

O Capítulo 6 apresenta os resultados referentes à modelagem dos sistemas de tráfego através da álgebra max-plus. Além da metodologia para modelagem e controle, diversos resultados de simulação são apresentados neste capítulo.

O Capítulo 7 apresenta as conclusões e algumas perspectivas futuras para a continuação deste trabalho.

Por fim, o Apêndice A contém uma lista de diversos termos e conceitos correntes da literatura de tráfego.

Capítulo 2

Coordenação Semafórica

Conforme visto no capítulo anterior, a sincronização entre semáforos de interseções vizinhas é um aspecto importante a considerar no controle de tráfego urbano, pois permite uma progressão de pelotões de veículos sem necessidade de parada em cada interseção, minimizando a poluição e gasto de combustível, além de aumentar o conforto dos motoristas.

Este capítulo apresenta o detalhamento do problema de coordenação e uma revisão das técnicas empregadas na solução do problema.

2.1 Coordenação entre Interseções

Quando interseções semaforizadas são localizadas nas proximidades de outras, o controle de cada interseção isoladamente interfere no comportamento de chegada de veículos nas outras interseções. Estas chegadas deixam de ser randômicas, ocorrendo na forma de pelotões nas interseções a jusante. Os pelotões são criados pela descarga da fila de veículos formada durante a indicação semafórica vermelha. Quando a distância entre interseções não é grande o suficiente para que ocorra dispersão, considera-se que o pelotão gerado na interseção a montante chega inalterado na interseção a jusante. Este comportamento motiva a tentativa de coordenar semáforos de forma a privilegiar a passagem destes pelotões, [Cervantes, 2005].

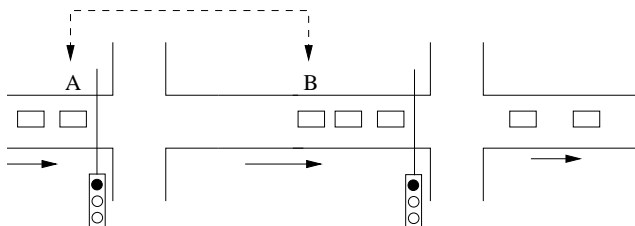


Figura 2.1: Situação desejada para coordenação entre interseções

A Figura 2.1 ilustra uma situação modelada em muitas técnicas de coordenação entre interseções. Deseja-se que o carro A alcance o carro B no exato instante que este começa a se mover, ou seja,

busca-se uma coordenação entre as interseções de maneira que os veículos que partem a montante encontrem o final da fila a jusante no exato momento em que ela começa a se movimentar, evitando assim uma parada. Uma variável importante nestas técnicas é a fila formada nas linhas de parada. Uma discussão sobre as filas é apresentada na seção 2.3.

Entretanto, em outras técnicas, busca-se a ocorrência do fenômeno conhecido como *onda verde*, onde a fila não é a principal variável. A *onda verde* acontece quando os semáforos são coordenados de maneira que um veículo que receba sinal verde em um extremo de uma arterial, trafegando em velocidade pré-determinada, possa percorrê-la até a outra extremidade sem parar em nenhum sinal vermelho durante o trajeto. Para que isto ocorra o tempo de ciclo tem que ser o mesmo em todas as interseções do trajeto e não podem existir filas longas nos cruzamentos a jusante.

As técnicas de coordenação podem ser classificadas quanto aos critérios utilizados para obter coordenação de semáforos. Baseado na literatura, é possível dividir as técnicas de coordenação em dois grandes grupos [Carlson, 2006]:

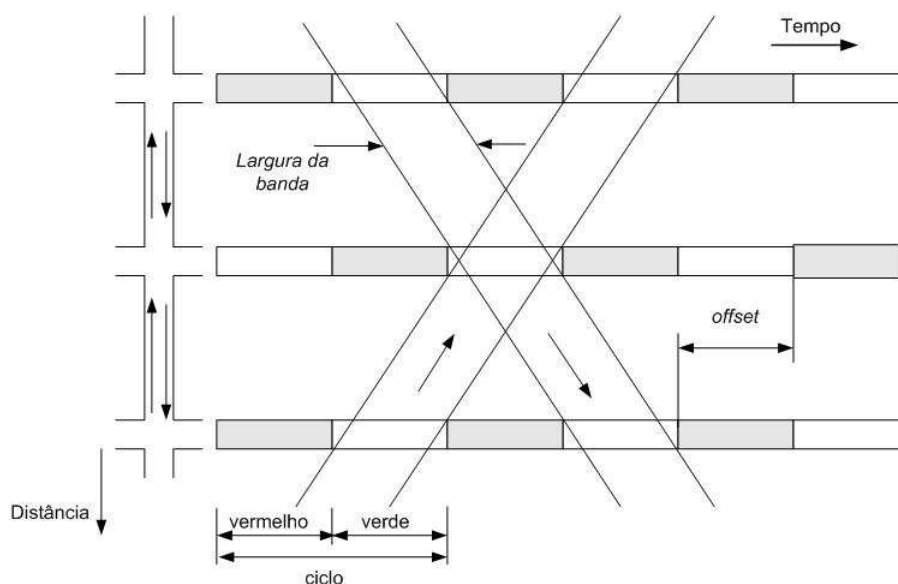
1. Técnicas que realizam maximização de largura de banda.
2. Técnicas que otimizam algum critério de tráfego, como atraso e número de paradas.

2.2 Técnicas de Coordenação

A distância temporal entre a linha imaginária do percurso do primeiro e do último veículo que passam desimpedidos por todos os semáforos de uma via arterial é denominada largura de banda de verde. Técnicas que procuram tirar o máximo proveito do tempo de verde no ciclo para geração de bandas são conhecidas por técnicas de maximização de banda de verde [Morgan e Little, 1964]. A Figura 2.2 apresenta um diagrama espaço-tempo para uma arterial. As linhas horizontais indicam os ciclos ao longo do tempo. As barras diagonais mostram a banda de verde em ambos os sentidos da arterial, limitada pelas trajetórias do primeiro e do último veículo que usufruem da banda. A inclinação da reta representa a velocidade dos veículos.

Com base em dados geométricos da via, tempo de ciclo e de porcentagens de verde, as técnicas de maximização da largura de banda obtêm valores de defasagem que proporcionam as chamadas ondas verdes. Trabalhos de destaque referente a maximização de banda verde para arteriais são os propostos por Morgan e Little [1964] e Little [1966].

O programa MAXBAND [Little et al., 1981] implementa a resolução de um programa inteiro misto como o desenvolvido por Little [1966], mas com adições à formulação original. MAXBAND obtém a coordenação que maximiza bandas em ambos os sentidos. A partir do MAXBAND foi desenvolvido o programa MULTIBAND [Gartner et al., 1990], que é uma proposta que procura combinar as vantagens das técnicas de maximização de banda e da otimização de critérios de tráfego simultaneamente em um modelo de otimização.



Fonte: Cervantes [2005]

Figura 2.2: Diagrama espaço-tempo

Em Carlson [2006] é apresentada uma estratégia de controle em tempo real que calcula defasagens para obtenção de larguras máximas de banda. Este trabalho mescla o método de Morgan e Little [1964] com a estratégia de controle em tempo real TUC (*Traffic-Responsive Urban Control*) [Diakaki et al., 2003]. A estratégia proposta no TUC implementa o controle de porcentagens de verde, ciclo e defasagem. O trabalho de Carlson [2006] aplica o método de Morgan e Little [1964] para calcular defasagens que proporcionem larguras máximas de banda.

O favorecimento da progressão de veículos por meio de ondas verdes obtidas através das técnicas de maximização de largura de banda é de fácil e intuitiva assimilação por operadores e motoristas, e é largamente utilizada. Diagramas espaço-tempo, como ilustrado na Figura 2.2, permitem a visualização do resultado. Estas técnicas utilizam poucos dados de entrada, quando são comparadas as técnicas que otimizam critérios de tráfego. Entretanto, o favorecimento da progressão na arterial não pode ser tal que prejudique em demasia as vias transversais. A adoção de planos obtidos por banda máxima é satisfatória quando o volume de tráfego está baixo, e de pouca utilidade quando está alto. Além disso, o desempenho de sistemas que empregam a maximização de banda é melhor quando poucos veículos fazem conversão das vias secundárias para arterial, ou seja, quando o fluxo é predominante na arterial [Carlson, 2006].

O segundo grupo de técnicas de coordenação contempla as técnicas que otimizam critérios de tráfego, por exemplo, técnicas que buscam minimizar o atraso veicular ou o número de paradas. O método mais difundido deste grupo é o TRANSYT [Robertson, 1969].

O critério atraso veicular acumulado está diretamente relacionado à formação de filas. Ele avalia o desempenho do tráfego através do número de veículos parados em fila. Assim, quanto maior a fila, maior o atraso. A formação de filas nas interseções ocorre em duas situações:

- durante o vermelho, quando os veículos são obrigados a parar; e
- quando a razão de chegadas é maior do que a capacidade de descarga da via durante o tempo de verde do semáforo.

O processo de cálculo do atraso depende do conhecimento da variável número de veículos em fila, e para quantificá-la é necessário conhecer o fluxo de veículos nas vias. Este fluxo pode ser obtido diretamente das contagens realizadas por sensores localizados nas vias ou manualmente por levantamento de dados históricos.

Nos algoritmos de tempo fixo e de controle atuado ¹ onde a variável *offset* (ver Apêndice A) é definida, o critério atraso veicular está relacionado com ela. Isto ocorre pois um ajuste sincronizado dos semáforos adjacentes pode proporcionar uma diminuição no tamanho da fila em uma via arterial.

O critério número de paradas é obtido pela avaliação da quantidade de todo tráfego que sofre atraso, ou seja, quando se tem veículos parados em fila existe contribuição para o número de paradas. As seguintes situações devem ser consideradas quando ocorre uma parada de veículo [Cervantes, 2005]:

- quando a fila está sob indicação semafórica vermelha: todos os veículos parados em fila contribuem para o número de paradas;
- quando os veículos encontram o final da fila e esta já recebeu indicação semafórica verde: os veículos que por ventura, pararem ao encontrar o final da fila ainda parada e os veículos que não conseguirem cruzar a interseção dentro da indicação verde, contribuem para o número de paradas;
- quando a fila existente é descarregada, mas o veículo recebe indicação semafórica vermelha ao chegar próximo a linha de parada: neste caso os veículos que não conseguem atravessar a interseção contribuem para o número de paradas.

Além dos critérios atraso veicular e número de paradas, é possível medir o consumo de combustível e poluição atmosférica. Entretanto, estes critérios são aspectos que surgem como consequência dos atrasos e paradas, quando os veículos são obrigados a desacelerar, parar e acelerar novamente. Além disso, os critérios de atraso e parada são utilizados algumas vezes em conjunto, e outras vezes apenas o atraso.

A próxima seção discute alguns modelos de filas utilizados nas técnicas que minimizam algum critério de tráfego.

¹Estratégias de controle atuado monitoram o tráfego, e com base nos dados coletados realizam, freqüentemente, procedimentos de otimização para ajustar os tempos semafóricos. No caso dos algoritmos de tempo fixo os tempos semafóricos não se adaptam às alterações do tráfego em tempo real.

2.3 Modelos de Filas

Como citado na seção anterior, dois critérios de tráfego que frequentemente são minimizados em algumas técnicas de coordenação de semáforos são o atraso veicular e o número de paradas. Os dois critérios são medidos em função da fila formada nas interseções.

A evolução determinística das filas pode ser descrita pela equação de estados apresentada a seguir:

$$x(t) = x(t-1) + a(t, t-1) - d(t, t-1) \quad (2.1)$$

onde $x(t)$ é a quantidade de veículos em fila no tempo t , $a(t, t-1)$ é o número de veículos que chegam no intervalo de tempo $(t, t-1)$ e $d(t, t-1)$ é o número de veículos que partem no intervalo de tempo $(t, t-1)$. Expressando em palavras, pode-se dizer que a fila, em um determinado intervalo de tempo, é igual ao número de veículos que existia na fila no tempo anterior somado ao número de veículos que chegam, menos o número de veículos que partem do ponto de referência utilizado.

A descarga de filas é realizada a uma taxa máxima igual ao fluxo de saturação da via. A descarga Q_s da fila pode ocorrer em três situações, [Cervantes, 2005]:

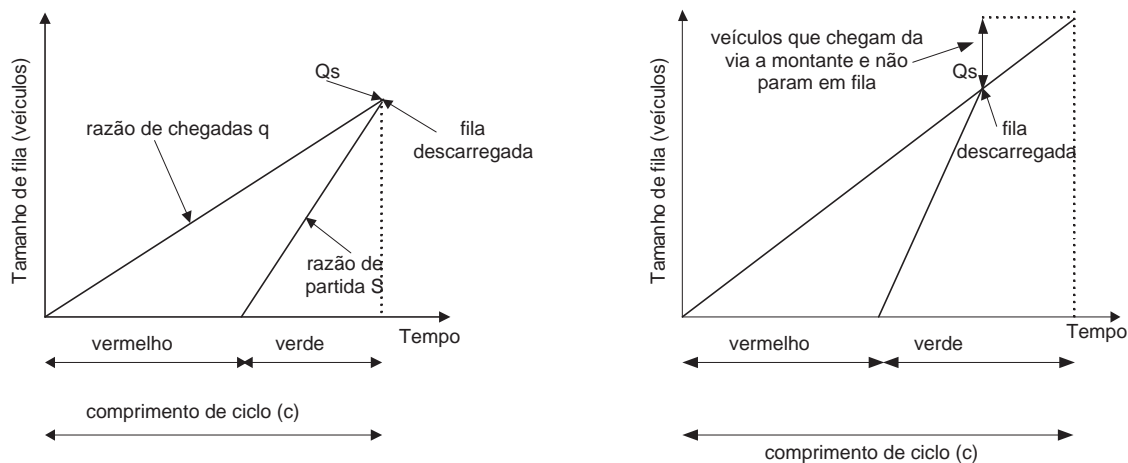
1. Quando a duração do tempo de verde é suficiente e justo para descarregar a fila formada durante o tempo de vermelho da via. Este é o caso saturado pois o fluxo veicular está no limite da capacidade da interseção, ou seja, a descarga Q_s ocorre até o final do tempo de verde. O gráfico à esquerda da Figura 2.3 ilustra esta situação.
2. Quando a descarga Q_s termina antes do término do tempo de verde o sistema é dito não-saturado, e neste caso, sobra tempo para descarregar o pelotão que chega da via a montante, como é possível verificar no gráfico à direita da Figura 2.3.
3. Uma terceira situação ocorre quando a descarga não é finalizada no tempo de verde, restando assim uma fila residual para o próximo tempo de vermelho. Este sistema é chamado sobre-saturado.

Os modelos mais utilizados para descrever a formação e descarga de filas são o modelo de fila vertical e o modelo de fila horizontal, e serão apresentados a seguir.

2.3.1 Modelo de Fila Vertical

Duas simplificações críticas são adicionadas aos modelos de fila para avaliar o atraso em interseções semaforizadas:

1. A hipótese que os veículos podem desacelerar e acelerar instantaneamente.
2. A hipótese que os veículos param em fila verticalmente, ou seja, os veículos atravessam todo o comprimento da via até chegarem na linha de parada, onde formam a fila.



Fonte: Cervantes [2005]

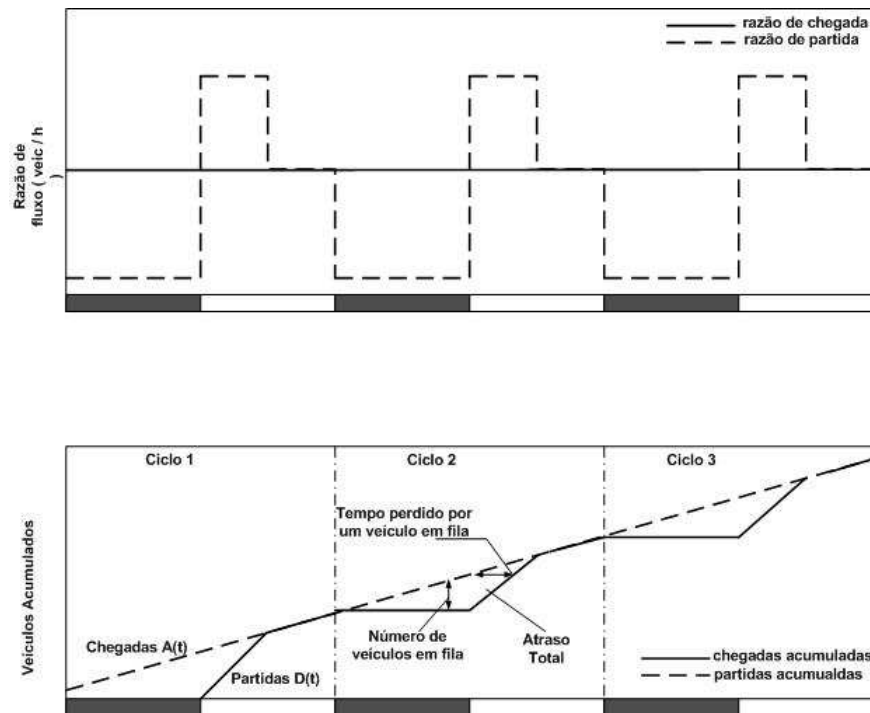
Figura 2.3: Formação e descarga de filas

A primeira hipótese implica na conversão dos atrasos de aceleração e desaceleração em atraso de parada. Desta maneira a diminuição da velocidade é considerada como se os veículos já estivessem parados. É preciso salientar que devido a hipótese de fila vertical, não é possível obter diretamente o comprimento da fila. Isto se deve ao fato que os veículos parecem parar mais tarde do que realmente aconteceria na prática.

O modelo de fila vertical assume uma razão de chegadas cumulativas e uma razão de partida inicialmente igual à taxa de descarga. Uma vez que a fila já foi descarregada a taxa de partida torna-se igual a taxa de chegada. Este fenômeno pode ser observado na Figura 2.4 [Cervantes, 2005], onde a curva de chegadas representa o número de veículos que chegariam na interseção se não sofressem parada no semáforo, enquanto a curva de partida representa o número de veículos que deixam a interseção. A distância vertical entre a curva de chegada e a curva de partida representa o número de veículos que não conseguem atravessar a interseção e portanto, param na linha de parada em fila. Por outro lado, a distância horizontal representa o tempo gasto por um veículo esperando na fila [Kang, 2000].

Através da Figura 2.4 é possível identificar três períodos distintos de evolução no tamanho da fila, durante um comprimento de ciclo:

1. O primeiro período corresponde a um intervalo durante o qual a curva de partida é horizontal. Neste caso a indicação semafórica está vermelha e o tráfego não pode atravessar a linha de parada, resultando no crescimento da fila.
2. Durante o segundo período a fila começa a ser descarregada com uma razão igual ao fluxo de saturação. Isto ocorre na primeira porção da fase de verde.
3. O último período só ocorre no caso não-saturado, onde todas as chegadas atravessam a linha de parada sem sofrer atraso. Neste caso a fila formada durante tempo de vermelho foi completamente dissipada.



Fonte: Cervantes [2005]

Figura 2.4: Chegadas e partidas de veículos acumulados para um caso ideal

O atraso total em um ciclo é estimado através do cálculo da área entre as curvas de chegada e de partida. Através da Figura 2.5 é possível observar que a fila atinge o valor máximo imediatamente antes do início do tempo de verde efetivo. O tempo requerido para descarregar totalmente a fila é dado como uma função da diferença entre a razão de chegada em que os veículos chegam ao final da fila e a razão de descarga enquanto eles cruzam a linha de parada [Cervantes, 2005].

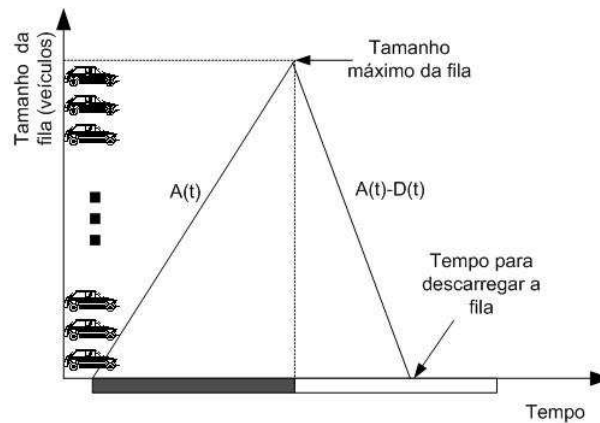
2.3.2 Modelo de Fila Horizontal

A Figura 2.6 representa o modelo de fila horizontal utilizado no algoritmo SCOOT [Hunt et al., 1982]. A Figura 2.6 foi adaptada por Cervantes [2005].

Um detector é localizado logo após a linha de parada da interseção a montante, e desta forma todos os veículos que chegam na via são detectados. Conhecendo-se o comprimento da via e a velocidade média de percurso dos veículos, torna-se possível determinar o tempo que os veículos vão gastar para chegar na linha de parada da interseção em análise (tempo τ na Figura 2.6). É necessário definir um tamanho médio para os veículos. Então são criadas duas variáveis:

- $B(t)$ que define o final da fila, e é dado em metros e valor inicial zero.
- $F(t)$ que define o início da fila, também dado em metros e valor inicial zero.

Quando o semáforo encontrar-se na indicação vermelha $B(t)$ começa a ter valores crescentes, pois os veículos começam a parar um atrás do outro. Depois $F(t)$ começa a crescer (para trás, em



Fonte: Cervantes [2005]

Figura 2.5: Modelo de fila vertical

distância) quando o semáforo abre e a fila começa a descarregar. Seus valores são crescentes até encontrar o final da fila, e então $B(t) = F(t)$, quando as duas são levadas a zero. Isto significa que os veículos que continuarem chegando não terão mais que parar em fila.

Através desta abordagem é possível modelar o comportamento de formação de filas e chegada de veículos na fila em diferentes pontos da via.

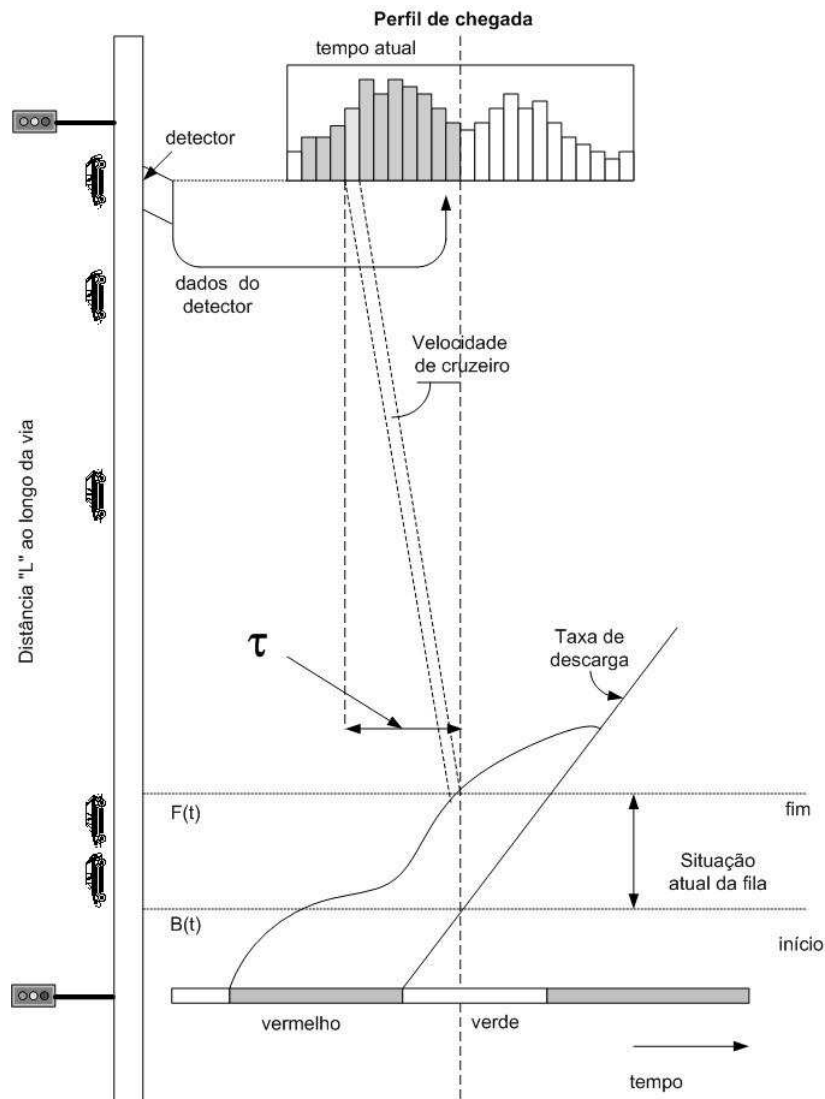
2.3.3 Comparação Entre os Modelos de Fila Vertical e Horizontal

Os modelos de filas verticais não representam o espaço físico ocupado pelos veículos, apenas retornam o número de veículos parados e seu atraso. Por outro lado, o modelo de fila horizontal permite representar o espaço físico, e portanto é indicado para modelar congestionamentos e bloqueios. Apesar destas diferenças os modelos podem ser comparados.

Os modelos de fila vertical e horizontal são equivalentes para o caso do sistema não saturado [Cervantes, 2005]. Ao observar a Figura 2.7 é possível observar que as linhas A, B, C, D e E, que representam as trajetórias de veículos que se aproximam da linha de parada com velocidade de cruzeiro, são as mesmas fora da região sombreada, e isto é esperado pois os dois modelos devem representar a fila. A principal diferença entre os modelos está no fato do modelo horizontal, ao modelar a ocupação espacial dos veículos, permitir a obtenção com maior precisão e de forma explícita da localização do último veículo em fila, dado importante para algumas técnicas de coordenação.

2.4 A Coordenação em Sistemas Integrados de Controle de Tráfego

Além da coordenação, outro problema importante no controle de tráfego é o cálculo dos tempos semafóricos. Propostas têm sido apresentadas na literatura para resolver de forma integrada os dois problemas de controle.

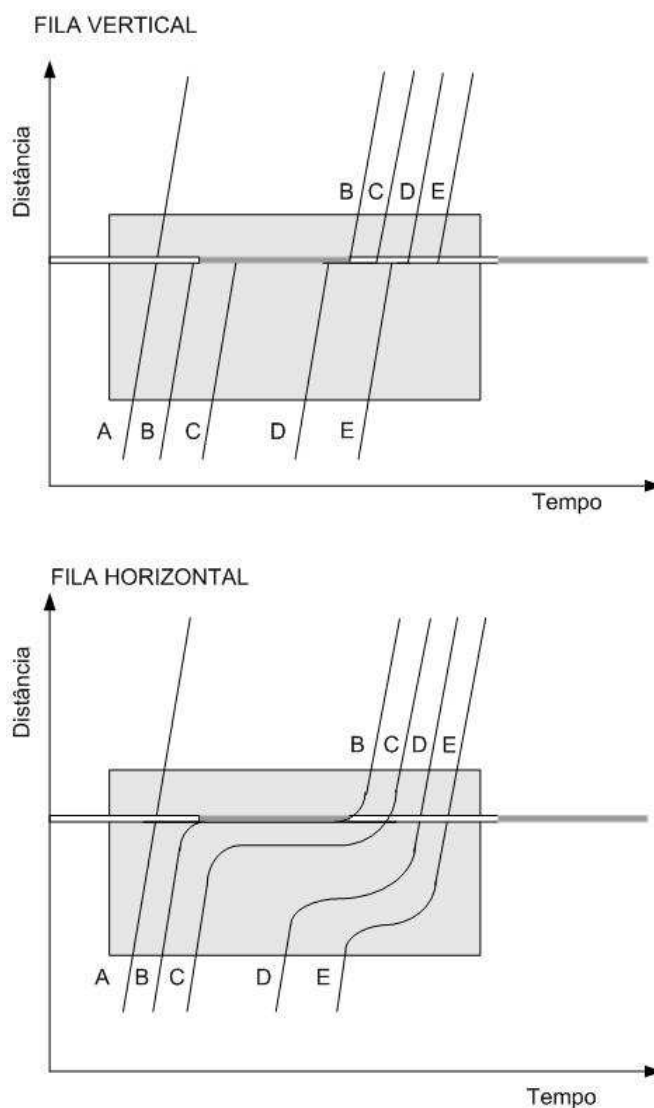


Fonte: Cervantes [2005]

Figura 2.6: Fila horizontal

Para embasar esta discussão, é importante definir os conceitos de ciclo, *offset* e *split* (para uma revisão dos conceitos de tráfego ver Apêndice A). Um ciclo corresponde ao intervalo de tempo entre o início do estágio principal de uma interseção e o instante em que ele se inicia novamente, tendo ocorrido outros estágios neste período de tempo. A diferença de tempo entre o início de estágios pré-determinados de duas interseções consecutivas é denominada *offset* e a divisão do tempo de verde de uma interseção entre seus estágios é definida pelo *split*.

Os sistemas de controle de tráfego são classificados como sistemas de controle de tempo fixo ou sistemas de controle em tempo real (também chamados sistemas atuados pelo tráfego). Planos de tempo fixo são baseados em fluxos de tráfego médios para um período de tempo em que eles estão operacionais. Entretanto, planos fixos não se adaptam em tempo real às alterações dos padrões de fluxo, pois não levam em conta a detecção de tráfego em tempo real.



Fonte: Cervantes [2005]

Figura 2.7: Comparação entre fila vertical e horizontal

O exemplo mais conhecido de sistema de controle de tempo fixo é o TRANSYT [Robertson, 1969]. A coordenação dos tempos semafóricos entre interseções é obtida através da otimização dos *offsets* entre interseções adjacentes, levando em conta os fluxos que ocorrem na malha viária. Os planos semafóricos fixos especificam o tamanho das fases, tempo de ciclo e *offset* para cada interseção sinalizada do sistema de controle de tráfego urbano.

Os sistemas de controle em tempo real surgiram para suprir as deficiências encontradas no controle a tempo fixo, como por exemplo o envelhecimento dos planos semafóricos. Os sistemas de controle em tempo real podem ser cíclicos ou acíclicos [Carlson, 2006].

Os sistemas cíclicos calculam os tempos semafóricos com base nos conceitos de ciclo, *offset* e *split*. O cálculo destes tempos são baseados em informações recebidas de detectores veiculares. Conhecendo-se o perfil do tráfego é possível ajustar os valores de *split* e *offset* a fim de obter

coordenação e minimizar algum critério de tráfego, como o tamanho das filas, o número de paradas ou atraso veicular. Exemplos de sistemas de controle em tempo real cíclicos são o SCOOT (*Split, Cycle, and Offset Optimisation Technique*) [Hunt et al., 1982, 1981], SCATS (*Sydney Co-ordinated Adaptive Traffic System*) [Lowrie, 1982] e TUC (*Traffic-responsive Urban Control*) [Diakaki et al., 2003; Dinopoulou et al., 2000].

Os sistemas acíclicos não se baseiam nestes conceitos e decidem a cada intervalo de controle (tipicamente em torno de 4 segundos) se o estado do semáforo deve ser mantido ou trocado, determinando assim que fases tem direito de passagem. Estes sistemas utilizam técnicas de otimização e predição em conjunto com a técnica de horizonte deslizante. O horizonte deslizante é construído utilizando um modelo de tráfego e medidas dos sensores. A cada período de controle, n períodos de controle futuros são simulados e uma árvore de decisão é construída. A simulação considera todas as possibilidades de mudança de sinal e as restrições de verdes mínimos e máximos. É realizada durante o período de controle vigente para que a decisão seja aplicada no período seguinte. A cada novo período o horizonte deslizante é reconstruído.

Exemplos de sistemas de controle acíclicos são o PRODYN [Farges et al., 1983], OPAC (*Optimization Policies for Adaptive Control*) [Gartner, 1983], CRONOS (*ContROl of Networks by Optimization of Switchovers*) [Boillot et al., 1992] e o ALLONS-D (*Adaptive Limited Lookahead Optimization of Network Signals – Decentralized Version*) [Porsche et al., 1996],

Conforme ressaltado neste capítulo, o problema tratado nesta tese é o da coordenação semafórica, e não do projeto dos tempos semafóricos. O objetivo é desenvolver ferramentas para a análise e síntese de coordenação. Estas ferramentas estão normalmente embutidas nos métodos cíclicos, e são importantes em processos iterativos para melhorar o ajuste do controle como um todo.

2.5 Conclusão

Neste capítulo discutiu-se alguns conceitos relacionados à coordenação semafórica. As técnicas de coordenação foram classificadas em dois grupos: as que minimizam algum critério de tráfego e as que maximizam a largura de banda. Os modelos de filas que podem ser utilizados pelas técnicas também foram apresentados. Além disso, alguns sistemas integrados de controle de tráfego também foram citados.

Como citado no Capítulo 1, os objetivos desta tese são investigar técnicas de análise e controle de sistemas a eventos discretos e sistemas híbridos para resolver o problema de coordenação semafórica. Primeiramente, os sistemas de tráfego urbano serão analisados como sistemas híbridos, entretanto, antes disso é necessário conhecer alguns conceitos referentes à modelagem e análise dos sistemas híbridos.

Uma introdução aos sistemas híbridos é apresentada no Capítulo 3, onde o formalismo de modelagem utilizado nesta tese, o autômato híbrido, também é discutido. Além disso, aspectos relacionados à verificação formal de propriedades também são abordados. Estes conceitos são necessários para entender a proposta de modelagem apresentada no Capítulo 4.

Capítulo 3

Sistemas Híbridos

No Capítulo 1 afirmou-se que os sistemas de tráfego urbano podem ser classificados como sistemas híbridos. Sistemas híbridos são sistemas que possuem tanto dinâmicas contínuas quanto dinâmicas discretas. As dinâmicas não apenas coexistem e sim interagem, mudanças ocorrendo tanto em resposta a eventos discretos como em resposta a dinâmicas descritas por equações contínuas no tempo [van der Schaft e Schumacher, 2000].

Os primeiros modelos desenvolvidos para representar os processos físicos tratavam separadamente as dinâmicas contínuas e discretas. Isto implicava modelos mais simples do sistema, porém não representavam adequadamente a natureza híbrida dos processos. É importante a utilização de modelos que representem os diferentes fenômenos físicos, contínuos e discretos, de forma integrada, representando o seu comportamento da forma mais completa.

Este capítulo tem o intuito de apresentar os sistemas híbridos. Será apresentado um exemplo, um breve histórico e o autômato híbrido, formalismo adotado neste trabalho para modelar os sistemas de tráfego urbano. Além disso, aspectos de verificação de sistemas híbridos também serão abordados.

3.1 Exemplo de um Sistema Híbrido

Um termostato para controlar a temperatura de um quarto é um bom exemplo para ilustrar um sistema híbrido [Dang, 2000]. Um termostato consiste de um aquecedor e um termômetro. O aquecedor fica ligado enquanto a temperatura está abaixo de um valor θ_M . Quando o termômetro detecta o valor θ_M o aquecedor é desligado, permanecendo assim até que a temperatura atinja o valor θ_m . É possível pensar que a temperatura do quarto e o termostato são sistemas dinâmicos, e cujo estado é definido pela temperatura do quarto x (que evolui continuamente) e o modo de operação do termostato (que evolui discretamente) pode estar ligado ou desligado. A evolução da temperatura pode ser descrita pelas seguintes equações:

$$\dot{x} = \begin{cases} f_1(x) = -x + 4 & \text{se o aquecedor está ligado,} \\ f_2(x) = -x & \text{caso contrário.} \end{cases}$$

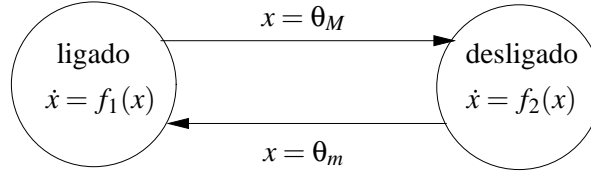


Figura 3.1: Modelo para o termostato

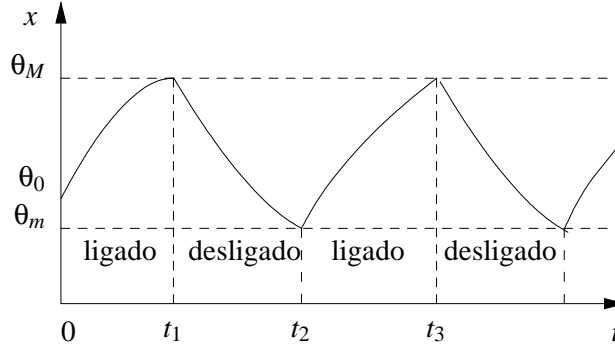


Figura 3.2: Trajetória para a temperatura do quarto

O termostato pode ser representado graficamente como um grafo dirigido, cujos vértices representam os dois modos de operação (ligado e desligado), conforme mostra a Figura 3.1. As condições de chaveamento entre os modos ligado e desligado são associadas aos arcos do grafo.

Um problema de verificação para este sistema seria, por exemplo, garantir que a temperatura sempre permanece em um intervalo desejado, que pode ser $m \leq x \leq M$.

Neste exemplo simples, para uma dada condição inicial $x(0) = \theta$ a solução da equação diferencial nos modos ligado e desligado são, respectivamente $x(t) = \theta e^{-t} + 4(1 - e^{-t})$ e $x(t) = \theta e^{-t}$.

É possível descrever um possível cenário do comportamento do sistema partindo de um estado inicial onde a temperatura é $x = \theta_0$ e o modo de operação do termostato é ligado. Supondo que a temperatura inicial está contida no intervalo desejado, isto é $m \leq \theta_0 \leq M$, o aquecedor inicialmente está ligado e a temperatura evolui segundo a seguinte equação:

$$x(t) = \theta_0 e^{-t} + 4(1 - e^{-t}).$$

Incrementando o tempo, a temperatura atinge o valor θ_M após o tempo t_1 , e o aquecedor é desligado. A temperatura passa a ser governada pela equação do modo desligado, que é descrita como:

$$x(t + t_1) = \theta_M e^{-t+t_1}.$$

A temperatura decresce até θ_m e o aquecedor é novamente ligado, como mostra a Figura 3.2. É possível perceber que a trajetória da temperatura alterna entre as duas fases correspondentes aos dois modos de operação do termostato.

Percebe-se que o termostato satisfaz a propriedade $m \leq x \leq M$ somente se os *thresholds*, ou eventos de limiar, satisfazem a seguinte condição: $\theta_m \geq m \wedge \theta_M \leq M$. Entretanto, o problema de verificação pode ser resolvido de maneira analítica somente quando as soluções para as equações diferenciais são conhecidas. Em casos mais gerais, simulações numéricas são utilizadas para obter uma aproximação do comportamento do sistema para um dado estado.

Para muitos sistemas, o estado da arte em técnicas de simulação permite que a solução aproximada seja tão próxima quanto se queira da solução exata. Entretanto, na prática as condições iniciais não são conhecidas de maneira exata, sabe-se apenas um intervalo onde elas estão contidas. Consequentemente, ao invés de ser considerada uma trajetória simples é necessário considerar um número infinito de trajetórias. Neste caso a abordagem de simulação não é adequada para verificação de propriedades de sistemas, e são necessários métodos rigorosos que possam caracterizar todos possíveis comportamentos do sistema. Outros exemplos de sistemas híbridos pode ser encontrados em Dang [2000]; Tomlin et al. [2003].

3.2 Estudos em Sistemas Híbridos

O estudo de sistemas híbridos tornou-se popular recentemente, mas os trabalhos pioneiros foram na década de 60. A primeira referência direta aos sistemas híbridos é o trabalho de Witsenhausen [1966], onde o autor trata um problema de controle ótimo para uma classe de sistemas dinâmicos de tempo contínuo. Este trabalho foi seguido por Pavlidis [1967], que estudou estabilidade de sistemas com impulsos utilizando funções de Lyapunov. O primeiro autor a defender o desenvolvimento de linguagens de simulação para sistemas que combinam dinâmicas contínuas com dinâmicas de eventos discretos foi Fahrland [1970]. Cellier [1979] iniciou a incorporação das dinâmicas discretas na simulação de sistemas contínuos, além de ser um dos primeiros a introduzir um conceito de estruturação para sistemas híbridos. Além destes, outros trabalhos de sistemas híbridos se seguiram: Johnson [1981] abordou modelos analíticos de processos com múltiplos estágios, Wimpey [1982] utilizou controladores de estados finitos para processos contínuos de tempo discreto. Ezzine e Haddad [1989] examinaram estabilidade, controlabilidade e observabilidade de uma classe restrita de sistemas híbridos, os sistemas lineares chaveados.

A partir do final dos anos 80 houve uma renovação em relação ao interesse pelos estudos sobre sistemas híbridos junto à comunidade de sistemas de controle. Isto se deveu ao desenvolvimento de pesquisas sobre controle de sistemas a eventos discretos, que ocorreu no final dos anos 80, ao surgimento do controle adaptivo nos anos 70 e 80 e ao renovado interesse em formulações de controle ótimo [Antsaklis, 2000].

Motivado pelo problema de modelagem de sistemas com histerese, Tavernini [1987] utilizou autômatos diferenciais para modelar sistemas híbridos e apresentou soluções para problemas com valor inicial e suas aproximações numéricas. Back et al. [1993] se basearam no modelo de Tavernini e introduziram um formalismo que modela chaveamentos e saltos autônomos e é adequado para simulação numérica. Nerode e Kohn [1993] introduziram uma abordagem da teoria de autômatos para

sistemas compostos de autômatos finitos e equações diferenciais que interagem entre si. Uma abordagem de sistemas dinâmicos a eventos discretos para sistemas híbridos foi introduzida por Antsaklis et al. [1993]; Brockett [1993] combinou equações diferenciais e fenômenos discretos para descrever sistemas de movimento. Os autômatos híbridos, que são uma extensão dos autômatos temporizados, foram introduzidos por Alur et al. [1993] e Alur e Dill [1994]. Benveniste [1996] propôs uma abordagem para modelagem de sistemas híbridos com ênfase na componibilidade dos seus componentes. Modelos de sistemas unificados, que capturam todos os fenômenos discretos que surgem em sistemas híbridos foram introduzidos por Branicky [1996].

Os sistemas híbridos compreendem uma área desafiadora tanto na ciência da computação quanto na teoria de controle. Segundo Miranda [2003] as abordagens para sistemas híbridos diferem em relação à ênfase ou à complexidade das dinâmicas contínuas e discretas que essas abordagens tratam e se enfatizam resultados de análise e síntese, ou somente análise ou verificação.

Em um extremo do espectro existem abordagens que buscam estender a teoria clássica de sistemas de tempo contínuo (descritos através de equações diferenciais ordinárias) de forma a incluir variáveis de tempo discreto ou variáveis que exibem saltos, ou ainda estender resultados para sistemas com chaveamentos. Estas abordagens são eficientes para tratar dinâmicas contínuas complexas e sua ênfase tem sido o estudo de estabilidade de sistemas com descontinuidades. Alguns trabalhos que podem ser citados são os de Lafferriere [1994] e Branicky et al. [1994].

No outro extremo do espectro estão as abordagens baseadas em modelos e métodos provenientes da ciência da computação e que visam estender as metodologias de verificação de sistemas a eventos discretos para o contexto de sistemas híbridos. Estas abordagens são capazes de tratar de dinâmicas discretas complexas, descritas por autômatos de estados finitos, e enfatizam resultados de análise (verificação) e simulação. Em Manna e Pnueli [1992] e Joseph [1988] são apresentados trabalhos que enfocam a verificação formal e em Antsaklis et al. [1995]; Grossman et al. [1993]; Maler et al. [1992] trabalhos que enfocam sistemas híbridos.

Existem, ainda muitas outras metodologias adicionais que combinam conceitos de sistemas de controle contínuo com conceitos de teoria de controle supervísório de sistemas a eventos discretos, de forma a obter resultados de análise e de síntese para sistemas híbridos, dentre eles os trabalhos de Cury e Krogh [1999]; Cury et al. [1998]; Raisch e O'Young [1998]. O trabalho de Leal e Cury [2004a,b] apresenta uma abordagem para controle modular de sistemas híbridos que sempre leva a uma solução ótima, onde a formulação do problema é realizada através da teoria de sistemas condição/evento e o problema é traduzido para um problema discreto, sendo a solução obtida através da teoria de sistemas a eventos discretos.

Existem diversos formalismos para representar os sistemas híbridos, entre eles os autômatos híbridos [Alur et al., 1996a], as redes de Petri híbridas [David e Alla, 2001], as redes de Petri diferenciais [Demongodin e Koussoulas, 1998], dentre outros. O modelo matemático adotado neste trabalho é o autômato híbrido, que é detalhado na seção 3.3.

Segundo Leal [2005], que propõe uma abordagem modular para resolver o problema de controle supervísório de sistemas híbridos, muitos autores vislumbram a utilização dos sistemas híbridos para

a resolução de problemas específicos. Algumas aplicações nas quais são utilizadas abordagens de sistemas híbridos são citadas a seguir:

- aplicações automotivas, tais como em sistemas de rodovias e automóveis inteligentes [Varaiya, 1993; Godbole e Lygeros, 1994; Puri e Varaiya, 1995b; Horowitz e Varaiya, 2000; Girard et al., 2001], e sistemas de tráfego urbano [Febbraro e Sacco, 2004];
- sistemas de segurança crítica [Livadas, 1997; Livadas e Lynch, 1998], tais como processos nucleares, sistemas de controle de tráfego aéreo [Tomlin et al., 1998] e sistemas de controle de aeronaves [Livadas et al., 2000; Seibel et al., 1999];
- sistemas de manufatura [Brandin, 1996; Pepyne e Cassandras, 2000; Villani, 2003];
- eletrônica de potência [Miranda, 2003].

3.3 Autômatos Híbridos

O autômato híbrido é um dos poucos formalismos que, em sua representação gráfica, consegue capturar, em um único modelo, as dinâmicas discretas e contínuas e a representação dos diferentes modos de operação de sistemas reativos [Miranda, 2003].

Um autômato híbrido [Alur et al., 1996a] é uma máquina de estados finitos generalizada que é equipada com variáveis contínuas, onde as dinâmicas para cada estado discreto são definidas através de equações diferenciais. Transições entre estados são habilitadas por condições nos valores destas variáveis contínuas.

3.3.1 Definição Formal

Definição 3.1 (Autômato Híbrido). *Um autômato híbrido é formalmente definido como uma tupla $\mathcal{A} = (X, Q, f, H, G, R)$, onde:*

- $X \subseteq \mathbb{R}^n$ é o espaço de estados contínuo. Elementos de X são escritos como $\mathbf{x} = (x_1, x_2, \dots, x_n)$;
- Q é o conjunto de estados discretos, também chamados de modos de controle ou locações;
- $f : Q \rightarrow (X \rightarrow \mathbb{R}^n)$ atribui um campo vetorial contínuo em X para cada estado discreto. Enquanto o autômato híbrido está no estado discreto q a evolução das variáveis contínuas é governada por uma equação diferencial $\dot{\mathbf{x}} = f(q)(\mathbf{x})$ ou uma inclusão diferencial $\dot{\mathbf{x}} \in f(q)(\mathbf{x})$;
- $H : Q \rightarrow 2^X$ são as condições de permanência, chamadas de invariantes. $H(q)$ é a condição que deve ser satisfeita pelas variáveis contínuas para permanecer no estado discreto q ;

- $G : (Q \times Q) \rightarrow 2^X$ são as guardas de transição que determinam as condições de chaveamento de um estado discreto para outro. Quando o autômato está no estado discreto q e $\mathbf{x} \in G(q, q')$ a transição de q para q' pode ocorrer;
- $R : (Q \times Q) \rightarrow (X \rightarrow 2^X)$ é o mapa de reset que atribui a cada transição uma função multivalor. $R(q, q')$ define como a variável contínua será alterada quando o autômato transitar do estado discreto q para q' .

O significado intuitivo da Definição 3.1 é que o conjunto de estados discretos Q denota todos os modos contínuos possíveis do autômato híbrido [Dang, 2000]. O sistema pode evoluir em um estado discreto q somente se o estado contínuo corrente está em $H(q)$. Estas condições de permanência podem ser definidas pelas restrições do sistema físico ou decisões de projeto do sistema. Enquanto o sistema está no estado discreto q , e $\mathbf{x} \in H(q)$, ele pode evoluir segundo as dinâmicas $f(q)$. No momento que ele alcança o ponto $\mathbf{x}' \in G(q, q')$ a transição do estado q para q' está habilitada e o sistema pode chavear para o estado discreto q' . Em q' as variáveis contínuas serão reiniciadas com novos valores de acordo com $R(q, q')$, que se tornam os estados iniciais para a evolução conforme $f(q')$. Convenciona-se que se não há transição entre q e q' então $G(q, q')$ é um conjunto vazio, e que, se o mapa *reset* $R(q, q)$ é a identidade, isto é, $R(q, q)(\mathbf{x}) = \mathbf{x}$ para todo $\mathbf{x} \in X$ então $G(q, q) = \emptyset$ porque uma transição de um estado discreto para ele mesmo sem alteração das variáveis contínuas não tem sentido.

Os autômatos podem ser representados graficamente como um grafo dirigido, onde os vértices representam os estados discretos e os arcos as transições. As condições invariantes e as equações diferenciais são escritas dentro dos vértices. As guardas e os *resets* são associados aos arcos. O autômato híbrido da Figura 3.3 ilustra esta representação, onde $Q = \{q_1, q_2\}$ e $x \in X$.

Para cada vértice do autômato existe um modo discreto, um campo vetorial ($f_1(x)$ para q_1 e $f_2(x)$ para q_2) e uma condição invariante (H_1 e H_2). Cada arco está rotulado com uma transição de guarda (G_{12} e G_{21}) e uma função de *reset* ($R_{12}(x)$ e $R_{21}(x)$), como mostra a Figura 3.3.

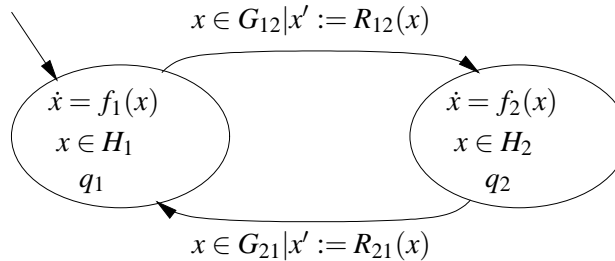


Figura 3.3: Representação gráfica de um autômato híbrido

A notação utilizada a partir deste momento será f_q para $f(q)$, $G_{qq'}$ para $G(q, q')$, G_q para $\bigcup_{q'} G_{qq'}$, H_q para $H(q)$, H para $\bigcup_q H_q$ e $R_{qq'}$ para $R(q, q')$.

O modelo de um sistema é composto pelo conjunto de todos os comportamentos que ele pode gerar. No caso de um sistema contínuo definido por uma equação diferencial, o modelo é definido

pelo conjunto de todas soluções de seu problema de valor inicial, em outras palavras, as trajetórias. Para o autômato híbrido \mathcal{A} considera-se o comportamento temporal sobre o espaço de estados híbrido $Q \times \mathcal{X}$. Um estado do autômato híbrido é representado por um par (q, \mathbf{x}) , e uma seqüência finita de estados (q, \mathbf{x}) configura uma trajetória. O par é formado por um estado discreto $q \in Q$ e \mathbf{x} um vetor de \mathcal{X} e pode ser alterado de duas maneiras:

1. Pela evolução contínua: os valores das variáveis contínuas são alterados de acordo com as dinâmicas f_q enquanto o estado discreto q permanece constante.
2. Pela evolução discreta: o sistema modifica os estados discretos realizando uma transição e possivelmente altera os valores das variáveis contínuas de acordo com a função *reset*.

Sejam $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ e $q, q' \in Q$ e, seja $\alpha : \mathbb{R}^+ \rightarrow \mathcal{X}$ a solução única de $\dot{\mathbf{x}} = f_q(\mathbf{x})$ com condição inicial $\alpha(0) = \mathbf{x}$. A dinâmica q está habilitada por \mathbf{x} para o tempo $t > 0$ se $\alpha(t') \in H_q$ para todo $t' \in [0, t]$. Isto é denotado por $\mathbf{x} \xrightarrow{q, t}$. O estado (q, \mathbf{x}') é alcançável partindo de (q, \mathbf{x}) pela dinâmica contínua f_q se existe um $t < \infty$ de modo que $\alpha(t) = \mathbf{x}'$ e $\alpha(t') \in H_q$ para cada $0 \leq t' \leq t$. Neste caso diz-se que \mathbf{x}' é q -alcançável partindo de \mathbf{x} e simboliza-se por $\mathbf{x} \xrightarrow{q, t} \mathbf{x}'$.

Um estado (q', \mathbf{x}') é alcançável a partir de (q, \mathbf{x}) através de uma transição discreta se $\mathbf{x} \in G_{q, q'}, \mathbf{x}' = R_{q, q'}(\mathbf{x})$ e $\mathbf{x}' \in H_{q'}$.

A Figura 3.4(a) traça um autômato híbrido de três estados, onde $\mathcal{X} = \mathbb{R}^2$. As guardas G_{12} , G_{23} são as regiões hachuradas da Figura 3.4(b), e a guarda G_{32} é o segmento de reta ab . Um esboço de uma trajetória do autômato da Figura 3.4(a), iniciando no estado (q_1, \mathbf{x}_0) está representado na Figura 3.4(b). O salto de \mathbf{x}_2 para \mathbf{x}_3 é devido a função *reset* R_{23} .

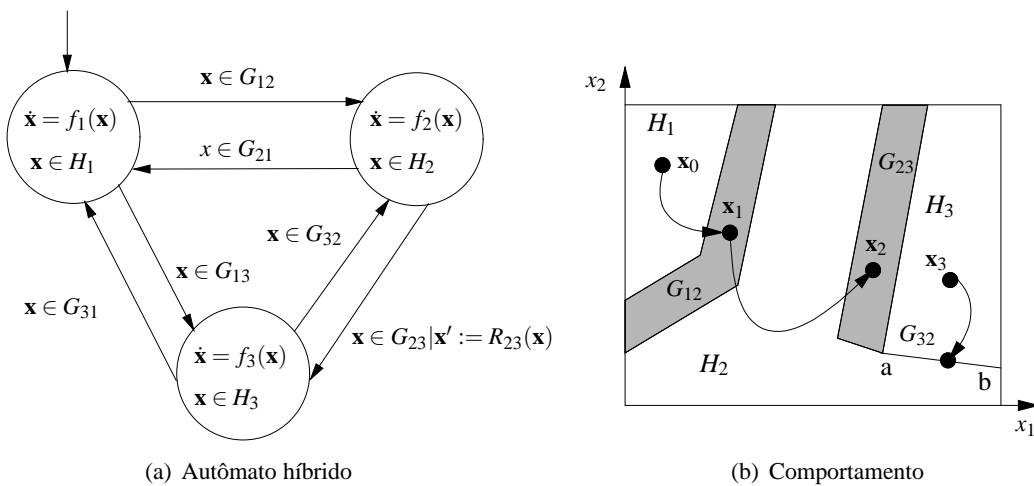


Figura 3.4: Autômato híbrido de três estados e seu comportamento

A utilização de invariantes e a possibilidade de chavear entre estados em tempo zero pode levar ao aparecimento de fenômenos impossíveis em modelos de sistemas contínuos bem comportados. Tais anomalias são conhecidas como bloqueio e comportamento zeno [Dang, 2000].

Uma trajetória é bloqueante se alcança um ponto (q, \mathbf{x}) do qual é impossível evoluir tanto por uma dinâmica contínua quanto por uma dinâmica discreta, isto é, $\mathbf{x} \notin H_q$ e $\mathbf{x} \notin G_{qq'}$ para todo $q, q' \in Q$. Um comportamento zeno é um comportamento constante por partes que tem um número infinito de transições em um intervalo de tempo limitado, ou seja, o sistema faz um número infinito de transições discretas em um tempo finito, e pode ser resultado da definição errada de invariantes. Os dois comportamentos são indesejados, e freqüentemente resultam de modelagens inadequadas do sistema.

3.3.2 Alcançabilidade em Autômatos Híbridos

O conjunto alcançável de um dado conjunto de estados \mathcal{F} por um autômato híbrido pode ser definido como o conjunto de todos os estados visitados pelas trajetórias iniciadas dos estados de \mathcal{F} . Uma trajetória de um autômato híbrido em um espaço contínuo de estados pode ser compreendida como uma seqüência de segmentos de trajetórias de dinâmicas contínuas.

Para obter os estados alcançáveis ou pesquisar o espaço de estados do sistema foram especificados operadores de alcançabilidade para um autômato híbrido \mathcal{A} . Como existem dois tipos de evoluções para um estado (q, \mathbf{x}) , evolução contínua e evolução discreta, foram definidos os operadores sucessor contínuo e sucessor discreto [Dang, 2000].

Definição 3.2 (Operador sucessor contínuo). *Dado um conjunto de estados \mathcal{F} composto de elementos (q, F) onde $q \in Q$ e $F \subseteq X$, o conjunto de sucessores contínuos de (q, F) , denotado por $\delta_c(q, F)$, é definido como*

$$\delta_c(q, F) = \{(q, \mathbf{x}') \mid \exists \mathbf{x} \in F \exists t > 0 ; \mathbf{x} \xrightarrow{q, t} \mathbf{x}'\}.$$

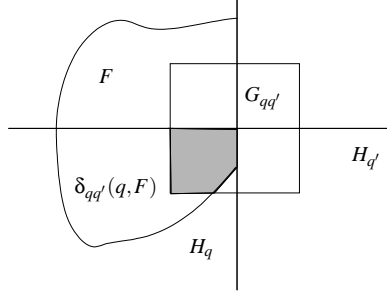
Lembrando que $\mathbf{x} \xrightarrow{q, t} \mathbf{x}'$ indica que \mathbf{x}' é q -alcançável de \mathbf{x} . Esta notação quer dizer que a trajetória da dinâmica f_q de \mathbf{x} para \mathbf{x}' está dentro de H_q .

Definição 3.3 (Operador sucessor discreto). *Dada uma transição de q para q' e um conjunto de estados (q, F) onde $q, q' \in Q$ e $F \subseteq X$, o conjunto de sucessores discretos de (q, F) , denotado por $\delta_{qq'}(q, F)$, é definido como*

$$\delta_{qq'}(q, F) = \{(q', \mathbf{x}') \mid \exists \mathbf{x} \in F \cap G_{qq'} \wedge \mathbf{x}' \in R_{qq'}(\mathbf{x}) \cap H_{q'}\}.$$

O conjunto $\delta_{qq'}(q, F)$ contém os estados (q', \mathbf{x}') , onde \mathbf{x}' são os pontos em $H_{q'}$, aplicando a relação reset $R_{qq'}$ para os pontos \mathbf{x} em F que satisfazem a guarda $G_{qq'}$.

A Figura 3.5 apresenta o resultado da aplicação deste operador. O conjunto $H_{q'}$ é o meio espaço abaixo da linha horizontal e a guarda $G_{qq'}$ é o quadrado. O sucessor discreto de (q, F) em relação à transição de q para q' pode se obtido pela intersecção de F com $G_{qq'}$ e depois com $H_{q'}$, que corresponde a região hachurada na Figura 3.5. Nesse caso assume-se que $R_{qq'} = I$, a identidade.

Figura 3.5: Sucessores discretos de F

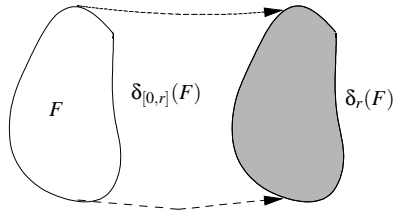
O conjunto de sucessores discretos de (q, F) , executando todas as transições habilitadas de q , denotado por $\delta_d(q, F)$, é definido como:

$$\delta_d(q, F) = \bigcup_{q' \in Q} \delta_{qq'}(q, F).$$

Uma vez obtidos os sucessores contínuos e discretos, através dos operadores δ_c e δ_d , o operador sucessor δ para \mathcal{F} é definido como o conjunto de estados alcançáveis de \mathcal{F} imediatamente após a evolução discreta, através das dinâmicas contínuas e transições discretas:

$$\delta(\mathcal{F}) = \delta_d(\delta_c(\mathcal{F})).$$

A notação δ_r é utilizada para representar os estados alcançáveis passado um tempo r . O conjunto alcançável de F é $\delta_{[0, \infty]}(q, F)$ denotado por $\delta(q, F)$. Este operador está ilustrado na Figura 3.6. A região preenchida representa $\delta_r(F)$ e o conjunto alcançável no intervalo de tempo $[0, r]$ é a região entre as duas linhas pontilhadas, a fronteira externa de F e o próprio $\delta_r(F)$.

Figura 3.6: Estados alcançáveis depois de um tempo r

O cálculo do conjunto de estados alcançáveis, a partir de um conjunto $\mathcal{F} \subseteq X$, $\delta(\mathcal{F})$, é a operação básica para qualquer procedimento de pesquisa no espaço de estados híbrido, especialmente verificação de propriedades.

O método de alcançabilidade para frente (*forward*) consiste em obter o espaço alcançável dado um conjunto inicial F de estados e calcular seus sucessores até que o conjunto de todos estados alcançáveis seja obtido. O algoritmo apresentado a seguir implementa o método de alcançabilidade para frente, e foi extraído de Dang [2000]:

procedure FORWARD


```

 $P^0 \leftarrow F$ 
repeat
   $k = 0, 1, 2, \dots$ 
   $P^{k+1} \leftarrow P^k \cup \delta_{[0,r]}(P^k)$ 
until  $P^{k+1} = P^k$ 
end procedure

```

O conjunto P^k , neste caso, é o conjunto de estados alcançáveis de F durante o intervalo $[0, kr]$ para $k = 0, 1, 2, \dots$ e dada uma etapa de tempo $r > 0$. O algoritmo termina quando nenhum novo estado alcançável é encontrado. A complexidade envolvida nos métodos de alcançabilidade é o cálculo do conjunto $\delta_{[0,r]}(P^k)$, ou seja, a região alcançável dentro do intervalo $[0, r]$, e tem sido motivo de muitas pesquisas na área de verificação de sistemas híbridos.

Para realizar o cálculo de estados predecessores, através do procedimento de alcançabilidade para trás (*backward*) utiliza-se o operador predecessor, tanto contínuo quanto discreto.

Definição 3.4 (Operador predecessor contínuo). *Dado um conjunto de estados (q, F) onde $q \in Q$ e $F \subseteq X$, o conjunto de predecessores contínuos de (q, F) , denotado por $\Delta_c(q, F)$, é definido como*

$$\Delta_c(q, F) = \{(q, \mathbf{x}') \mid \exists \mathbf{x} \in F \exists t > 0; \mathbf{x}' \xrightarrow{q,t} \mathbf{x}\}.$$

A Figura 3.7(a) ilustra o operador predecessor contínuo.

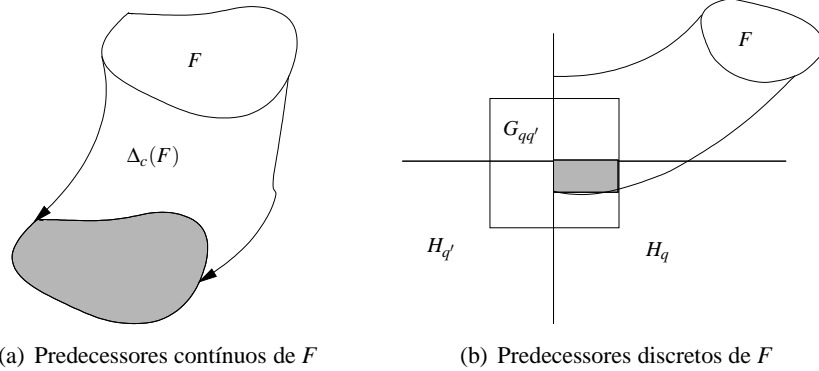


Figura 3.7: Operadores predecessores contínuos e discretos

Definição 3.5 (Operador predecessor discreto). *Dada uma transição $q, q' \in Q$ e um conjunto de estados (q, F) , onde $F \subseteq X$, o conjunto de predecessores discretos de (q, F) , denotado por $\Delta_{q'q}(q, F)$ é definido como*

$$\Delta_{q'q}(q, F) = \{(q', \mathbf{x}') \mid \exists \mathbf{x} \in F \cap R_{q'q}(\mathbf{x}') \wedge \mathbf{x}' \in H_{q'} \cap G_{q'q}\}.$$

O conjunto de predecessores discretos de (q, F) executando todas as transições habilitadas de q , denotado por $\Delta_d(q, F)$ é definido como:

$$\Delta_d(q, F) = \bigcup_{q' \in Q} \Delta_{q'q}(q, F).$$

Este operador está ilustrado na região hachurada da Figura 3.7(b), onde também se assume que $R_{qq'} = I$.

A escolha do método de alcançabilidade para frente ou para trás depende do tipo de estudo a ser realizado ou do modelo de sistema a ser tratado. O método de alcançabilidade para trás é normalmente utilizado quando se deseja verificar se, a partir de uma dada condição (representada por uma região no espaço de estados), um dado conjunto de condições iniciais do sistema é a sua origem [Miranda, 2003].

3.4 Verificação de Sistemas Híbridos

Sendo os sistemas híbridos freqüentemente sistemas complexos, sua análise automática torna-se desejável. A verificação de propriedades é a tarefa de determinar se o sistema está construído de acordo com uma dada especificação e, para realizá-la, o projetista pode utilizar técnicas matemáticas para modelagem e análise de sistemas. O procedimento mais importante para verificação formal e estudo do comportamento de sistemas híbridos é a análise de alcançabilidade. Esta técnica consiste no cálculo de todos os estados que podem ser alcançados por quaisquer trajetórias do sistema, dado um conjunto de condições iniciais.

Segundo Miranda [2003] e Dang [2000], calcular os estados alcançáveis para sistemas híbridos tem sido considerado um dos principais problemas em pesquisas para estes tipos de sistemas, uma vez que a obtenção do conjunto de estados alcançáveis constitui-se num dos pré-requisitos para utilização de metodologias de verificação algorítmica de propriedades de sistemas e é realizado sobre conjuntos no espaço \mathbb{R}^n , difíceis de serem calculados exatamente.

Diversos métodos de aproximação para o conjunto de estados alcançáveis do sistema tem sido propostos na literatura, assim como diversas abordagens para verificação de propriedades dos sistemas híbridos.

3.4.1 Métodos de Verificação Formal

Os dois principais métodos de verificação de propriedades de um sistema são a verificação automática (*model checking*) [McMillan, 1993; Alur et al., 1990] e o sistema de prova dedutiva [Manna e Pnueli, 1995]. A verificação automática é também conhecida como abordagem algorítmica de verificação.

A tarefa principal de um algoritmo de verificação automática é, recursivamente, calcular o conjunto de estados que satisfazem uma determinada propriedade. Este tipo de técnica exige um modelo

finito para representação do sistema, ou seja, o sistema a ser verificado representado por um modelo com número finito de possíveis estados alcançáveis e transições entre estes estados. A partir do espaço de estados deste modelo, é realizada uma busca exaustiva para comprovabilidade de uma dada especificação [Kowalewski, 2002].

Para sistemas híbridos, isto é realizado através de um procedimento comumente chamado de verificação simbólica, uma vez que, sobre um espaço de estados infinito, é necessário uma forma simbólica de representar conjuntos de estados e operações sobre eles, ou seja, utilizar uma forma de enumeração explícita para representar o espaço de estados [Miranda, 2003].

O sistema de prova dedutiva ou método dedutivo é o processo de encontrar uma prova de uma propriedade, a partir do uso de axiomas do sistema. As etapas dos processos de prova envolvem axiomas e regras, definições derivadas e lemas intermediários. Esta abordagem normalmente é mais abrangente que a abordagem algorítmica, mas ela não é automática e necessita da interferência da inteligência humana para encontrar as provas. Alguns trabalhos que abordam prova dedutiva são os de Manna e Pnueli [1995]; Manna et al. [1996].

3.4.2 Abordagens Algorítmicas de Verificação

A abordagem padrão para verificação de sistemas híbridos é a construção de uma bissimulação de estados finitos do sistema de transição de estados infinitos [Chutinan e Krogh, 2003]. Bissimulações são construídas usando partições finitas do espaço de estados original, levando ao chamado sistema de transição quociente (QTS). A dificuldade encontrada é que bissimulações de estado finito só existem para sistemas híbridos com dinâmicas contínuas triviais. Em geral, bissimulações finitas não existem, o que significa que o problema de verificação para sistemas híbridos normalmente é indecidível. Entretanto, o sistema de transição quociente computado para qualquer partição do espaço de estados do sistema de transição é uma simulação do sistema de transição. Isto significa que uma especificação universal (isto é, uma especificação que deve ser verdadeira para todas possíveis trajetórias) é verdade para o QTS, e é também verdade para o sistema de transição de estados infinitos (isto é, o sistema híbrido). Logo, mesmo com os problemas de indecidibilidade em geral, é possível verificar certas especificações. Não é possível, porém, prever se uma dada especificação será ou não verificável.

No trabalho de Dang [2000] as técnicas de verificação de sistemas híbridos são divididas em duas abordagens: direta e indireta. A abordagem direta trabalha diretamente no espaço de estados contínuo do sistema. Nesta abordagem é necessário encontrar uma aproximação para regiões do espaço de estados, devido à dificuldade de computar exatamente os conjuntos alcançáveis dos sistemas híbridos. Exemplos de algoritmos de verificação desta abordagem foram implementados nas ferramentas *KRONOS* [Yovine, 1997], *UPPAAL* [Bengtsson et al., 1998] e *HyTech* [Henzinger et al., 1997].

A abordagem indireta, por outro lado, reduz primeiro o sistema, através de uma abstração para um autômato de estados finitos. O processo de abstração consiste em encontrar uma partição finita do espaço de estados tal que a alcançabilidade entre as regiões de partição seja descrita pela relação de transição do autômato (detalhes em Alur et al. [2000]). Se a abstração pode ser construída, a

verificação pode ser executada sobre o sistema abstraído com a garantia de término, utilizando procedimentos normalmente utilizados para sistemas de estados finitos. Entretanto, como citado anteriormente, uma abstração finita existe apenas para uma classe restrita de sistemas híbridos, tanto com dinâmicas contínuas simples [Alur e Dill, 1994; Alur et al., 1993; Nicolin et al., 1993] como com dinâmicas discretas simples [Lafferriere et al., 1999]. Mesmo quando uma dada classe de sistemas admite um quociente finito, existe a dificuldade de calculá-lo efetivamente, pois é necessário calcular os sucessores de todas as regiões na partição, constituindo-se uma tarefa tão difícil quanto o problema de verificação. Contudo, deve ser notado que esta abordagem é importante para prova de corretude e término de algoritmos de verificação para algumas classes de sistemas [Alur e Dill, 1994; Henzinger, 1995] e permite verificar propriedades gerais de lógica temporal.

Alternativamente, outros trabalhos consideram aproximações discretas, isto é, ao invés de um quociente finito exato procura-se um sistema discreto finito, cujo comportamento inclui todos comportamentos do sistema original. No trabalho de Chutinan e Krogh [1999], aproximações discretas são obtidas por refinamentos sucessivos das partições de estado utilizando um método de alcançabilidade aproximado para dinâmicas contínuas. O método é semelhante ao apresentado em Dang [2000].

Além da abordagem direta e indireta é possível identificar uma classe mista, que consiste em derivar do sistema original um sistema aproximado para o qual são desenvolvidos ferramentas e algoritmos de verificação [Puri e Varaiya, 1995a; Stursberg e Kowalewski, 1999]. O sistema aproximado pode ser gerado através de uma sobreaproximação das dinâmicas contínuas complexas com dinâmicas simples (como inclinação constante e inclusões retangulares) baseadas na discretização do espaço de estado contínuo. O principal obstáculo desta abordagem é que, além do esforço da abstração inicial das dinâmicas contínuas com respeito à precisão desejada, o tamanho do sistema resultante pode ser proibitivamente grande para algoritmos de verificação.

Além das abordagens algorítmicas, encontra-se na literatura trabalhos sobre abordagem dedutiva (citado na seção 3.4.1) e abordagem de controle ótimo [Puri e Varaiya, 1995b] para verificação de sistemas híbridos.

3.5 Ferramentas

Diversas ferramentas computacionais foram desenvolvidas, tanto comerciais quanto acadêmicas, com o intuito de realizar verificação formal de sistemas híbridos. Os trabalhos de Silva et al. [2001], Leal [2005] e Alur et al. [2006] apresentam diversas ferramentas de modelagem, simulação e verificação de sistemas híbridos. Algumas delas serão apresentadas resumidamente a seguir:

A. UPPAAL

A ferramenta *UPPAAL* [Bengtsson et al., 1998] é um ambiente para modelar, simular e verificar sistemas representados como redes de autômatos temporizados. Os autômatos são definidos, de acordo com Alur e Dill [1994], com uma extensão para as variáveis de dados e mecanismos de

sincronização para modelar a comunicação de autômatos separados. A ferramenta pode analisar propriedades simples de vivacidade assim como propriedades de alcançabilidade.

UPPAAL foi desenvolvida conjuntamente na Universidade de Uppsala, na Suécia e na Universidade de Aalborg, Dinamarca. Detalhes sobre a ferramenta podem ser encontrados em <http://www.uppaal.com>.

B. HyTech

A ferramenta *HyTech* (*HYbrid TECHNOlogy Tool*) [Henzinger et al., 1997] verifica especificações definidas como expressões de lógica temporal para uma classe de autômatos híbridos utilizando um *model checking* no espaço de estados contínuo. Se a verificação falha é gerado um diagnóstico sobre o erro. O *HyTech* só modela sistemas com dinâmicas $A\dot{x} \leq B$, caracterizando os chamados autômatos híbridos lineares. Além disso é possível realizar composição paralela de processos e análise parametrizada.

O *HyTech* foi desenvolvido no EECS (Electrical Engineering and Computer Sciences Department) da Universidade da Califórnia (UC Berkeley), e outros detalhes podem ser encontrados em <http://www-cad.eecs.berkeley.edu/~tah/HyTech>.

C. d/dt

A ferramenta *d/dt* [Dang, 2000] realiza verificação e síntese de controladores para sistemas híbridos. Ela executa análise de alcançabilidade para sistemas híbridos com dinâmicas lineares, onde os conjuntos alcançáveis são representados como coleções de poliedros retangulares ortogonais. Estes conjuntos são computados através do método denominado *face lifting* [Dang e Maler, 1998], que calcula sobreaproximações eficientes. Além disso, *d/dt* permite trabalhar com equações dinâmicas com entrada incerta, isto é, modelos do tipo $\dot{x} = Ax + Bu, u \in U$.

A ferramenta foi desenvolvida no VERIMAG por Eugene Asarin, Oded Maler e Thao Dang, e outras informações podem ser obtidas em <http://www-verimag.imag.fr/~tdang>.

D. CheckMate

O *CheckMate* é uma ferramenta computacional baseada no *MATLAB/Simulink* para modelagem, simulação e verificação formal de sistemas híbridos, e foi desenvolvida na Universidade Carnegie Mellon. Modelos são construídos usando a interface gráfica *Simulink* e ela possibilita a verificação de sistemas híbridos através da obtenção de aproximações para seu modelo discreto [Chutinan, 1999].

Detalhes sobre a ferramenta podem ser obtidos em <http://www.ece.cmu.edu/~webk/checkmate>.

E. PHAVer

A ferramenta *PHAVer* (Polyhedral Hybrid Automaton VERifyer) realiza verificação formal de propriedades em sistemas híbridos modelados por intermédio de autômatos híbridos poliedrais [Frehse, 2005].

Informações sobre esta ferramenta podem ser encontradas em http://www-verimag.imag.fr/~frehse/phaver_web/

Além destas ferramentas apresentadas existem muitas outras, e informações sobre elas podem ser encontradas em <http://wiki.grasp.upenn.edu/~graspd/doc/wiki/hst>.

3.6 Conclusão

Ao longo deste capítulo os sistemas híbridos foram apresentados, sendo explorado um exemplo ilustrativo e um breve histórico. Além disso, o formalismo adotado neste trabalho, o autômato híbrido, foi estudado. Por fim, aspectos de verificação foram analisados, sendo apresentada uma breve revisão dos métodos e técnicas de verificação existentes na literatura juntamente com algumas ferramentas computacionais.

Os sistemas de tráfego urbano são classificados como sistemas híbridos, onde existem atividades contínuas devido ao fato das filas serem modeladas como fluxos contínuos. As dinâmicas discretas aparecem, dentre outras, em razão do chaveamento dos sinais semafóricos. O autômato híbrido, formalismo utilizado para modelar os sistemas de tráfego, permite tanto simular quanto verificar formalmente propriedades do sistema.

No próximo capítulo é apresentado um método para modelagem dos sistemas de tráfego urbano, buscando sempre realizar coordenação semafórica. A análise do comportamento das filas geradas no sistema é realizada através de simulação dos modelos obtidos e verificação de propriedades.

Capítulo 4

Modelagem e Análise de Sistemas de Tráfego como Sistemas Híbridos

Conforme citado no Capítulo 1, investigar o uso de sistemas híbridos para modelar coordenação semafórica em sistemas de tráfego urbano pode ser interessante pois os modelos obtidos permitem realizar uma análise do comportamento do sistema modelado. A análise pode ser realizada com base nos dados gerados através de simulações, e, quando possível, nas informações obtidas através da verificação formal de propriedades.

Ao longo deste capítulo os sistemas de tráfego são estudados como sistemas híbridos. Um método de modelagem baseado em autômatos híbridos é apresentado, e os modelos obtidos são implementados na ferramenta *CheckMate*. Ao final, resultados de simulação e verificação são discutidos.

4.1 Trabalhos Relacionados

Um sistema de tráfego urbano pode ser considerado um sistema híbrido, incluindo componentes dirigidos por eventos discretos e componentes contínuos no tempo. Para o caso dos sistemas de tráfego urbano, as dinâmicas discretas dos semáforos influenciam na definição das dinâmicas contínuas dos fluxos de tráfego.

Alguns trabalhos da literatura tem utilizado esta abordagem para representar os sistemas de tráfego urbano. Em Febbraro et al. [2003] um modelo híbrido para um sistema CTU (controle de tráfego urbano) é proposto. O objetivo do trabalho, resumidamente, é otimizar a performance de veículos especiais (como ambulância, veículos policiais e dos bombeiros) através da coordenação semafórica.

O modelo híbrido é desenvolvido para interseção sinalizada, onde as dinâmicas contínuas dos fluxos veiculares são naturalmente influenciadas pelas dinâmicas discretas dos semáforos, as quais definem se um fluxo tem ou não permissão de passagem. Além disso, o comportamento do tráfego é descrito através do modelo macroscópico de segunda ordem conhecido como modelo de Payne

[Payne, 1971]. As equações que descrevem o modelo podem ser encontradas em Febbraro et al. [2003].

O problema de coordenação apresentado em Febbraro et al. [2003] aborda a idéia que a rede de tráfego pode ser analisada como um grafo valorado (pesos são atribuídos aos arcos), onde cada nó representa uma interseção sinalizada e cada arco representa um *link* (uma via que liga duas interseções). É suposto, sempre que possível, que um veículo alcança um destino particular partindo de uma origem particular. Se o tempo de viagem em cada *link*, os tempos de espera devido aos semáforos em vermelho e os tempos de espera em filas são bem conhecidos a solução do problema de controle é baseado na busca de menor caminho em um grafo.

A estrutura de controle adotada é baseada em controladores locais e controladores de prioridade. Cada interseção é equipada com um controlador local e um controlador de prioridade. Os controladores locais atuam nas interseções como se elas fossem uma interseção isolada, onde o objetivo é minimizar as filas nas direções de entrada. O controlador de prioridades gerencia as prioridades dos veículos especiais, que possuem a prioridade mais alta, e podem forçar o controlador local a adequar as temporizações a fim de privilegiar os veículos especiais. Os controladores de prioridade são gerenciados por um supervisor que informa quando um veículo especial está se aproximando da interseção. Além disso, o cálculo do caminho ótimo entre a origem e destino do veículo especial é calculado pelo supervisor, que também comunica ao veículo qual é este caminho.

Entretanto, os grafos não são os únicos formalismos de modelagem utilizados na literatura. Nos trabalhos de Febbraro et al. [2001] e Febbraro e Sacco [2004] os sistemas são modelados através das redes de Petri híbridas.

As redes de Petri [Murata, 1989] são uma metodologia para modelar sistemas a eventos discretos, e estão aptas a capturar as relações de procedência e interações entre os eventos assíncronos e concorrentes típicos destes sistemas. Como resultado as redes de Petri não possuem vantagens apenas na capacidade de modelagem e auxílio das propriedades estruturais, mas são ferramentas poderosas de projeto, análise e controle. A idéia de aplicar as capacidades das redes de Petri na modelagem e controle de sistemas de tráfego foi introduzida por DiCesare et al. [1994], onde a representação da rede de tráfego foi realizada através de redes de Petri coloridas.

As redes de Petri híbridas [Alla e David, 2001] são uma combinação da rede de Petri clássica e uma rede de Petri contínua, a versão fluido de uma rede de Petri temporizada. A idéia básica é explorar as capacidades das redes de Petri citadas anteriormente para adequadamente representar e gerenciar os aspectos de concorrência e paralelismo que caracterizam os componentes de uma rede de tráfego urbano, regidos por dinâmicas contínuas e discretas.

O trabalho de Febbraro e Sacco [2004] apresenta uma definição para uma rede de transporte urbano. As interseções sinalizadas são os elementos da rede que conectam as diversas vias, e consistem na área física ocupada pelos veículos. Os fluxos dos veículos que circulam nas vias são a componente contínua do modelo híbrido proposto por Febbraro e Sacco [2004], enquanto o comportamento dos semáforos correspondem a componente discreta. A abordagem proposta é modular, onde cada interseção é modelada através de uma rede de Petri híbrida. A parte contínua da rede de Petri modela

os fluxos veiculares e a parte discreta as trocas semafóricas. O acoplamento de todas as interseções da rede de transporte urbano é dado por redes de Petri contínuas que modelam as vias adjacentes. O trabalho apresenta ainda alguns resultados experimentais para mostrar a eficácia do modelo. Além disso aborda aspectos relacionados a otimização da performance e controle.

Outro trabalho que também modela o sistema de tráfego como um sistema híbrido é o proposto por Lei e Ozguner [2001]. Este trabalho apresenta um controlador híbrido para áreas urbanas. Primeiramente o problema de controle para uma interseção isolada é analisado. Neste trabalho as filas nas vias são modeladas como a variação dos fluxos de entrada e saída de veículos, e considera-se que o tempo de ciclo, fase e *offset* são fixos.

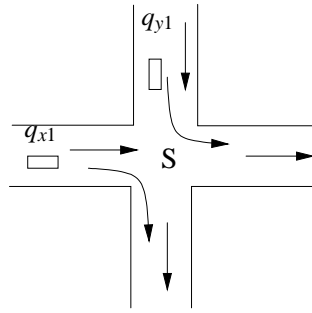


Figura 4.1: Interseção isolada

As dinâmicas propostas para a interseção mostrada na Figura 4.1 são:

$$\text{verde}_{x1} = \begin{cases} \dot{q}_{x1} = u_{x1}^i - u_{x1}^o \\ \dot{q}_{y1} = u_{y1}^i \end{cases}$$

$$\text{vermelho}_{x1} = \begin{cases} \dot{q}_{x1} = u_{x1}^i \\ \dot{q}_{y1} = u_{y1}^i - u_{y1}^o \end{cases}$$

onde u^i e u^o são respectivamente os fluxos de entrada e saída de veículos e q_{x1} e q_{y1} as variáveis contínuas que representam as filas nas direções x e y .

Um controlador híbrido é proposto, e isto é feito com base nas diferentes condições de tráfego de uma interseção isolada. Para estas interseções o controlador híbrido tem dois estados, um para a situação onde o sistema está saturado e outro para situação não saturado. Então, o método é estendido para o caso onde existem múltiplas interseções, sendo que são consideradas informações de tráfego de toda rede. Por fim, um controlador híbrido descentralizado com 5 estados é proposto, e ele permite manusear com diferentes combinações de situações de tráfego da rede.

Além dos trabalhos abordados aqui podemos citar os trabalhos de Gallego et al. [1996], Tolba et al. [2001], entre outros. A próxima seção introduz a proposta de modelagem para os sistemas de tráfego urbano desenvolvida nesta tese, onde o formalismo adotado é o autômato híbrido.

4.2 Modelagem em Alto Nível

Constantemente buscam-se alternativas para a modelagem de sistemas de tráfego urbano. Esta seção apresenta uma proposta de modelagem para estes sistemas. O objetivo da metodologia é encontrar um modelo geral do sistema, e o resultado obtido é um autômato híbrido [Alur et al., 1996a] que o representa. Inicialmente se deseja obter um autômato que represente o sistema num primeiro nível de abstração, e para tal é necessário seguir sete passos, que são apresentados a seguir.

Passo 1: Identificação da topologia

O primeiro passo é a definição da topologia do sistema; é necessário saber como as vias e as interseções sinalizadas que as conectam são representadas.

Cada interseção é vista como um nó, e conectado a este nó existem diversos *links*. Um *link* corresponde a uma via que se conecta à interseção sinalizada.

Existem fluxos veiculares associados aos *links*, e eles são definidos como:

- u_j^i : fluxo de entrada no *link* j , convencionado como sendo um fluxo no sentido do nó;
- u_j^o : fluxo de saída no *link* j , convencionado como sendo um fluxo entrando no nó;
- u_j^l : fluxo de entrada no *link* j , convencionado como sendo um fluxo saindo do nó;

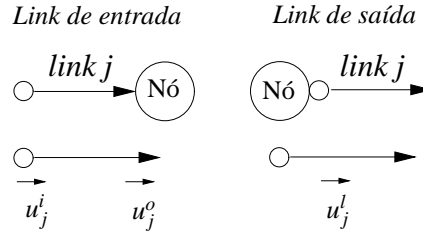
Os *links* podem ser classificados como:

- *link* de entrada: diz-se que um *link* j é um *link* de entrada quando existem um fluxo de entrada u_j^i e um fluxo de saída u_j^o na direção do nó associados a ele;
- *link* de saída: diz-se que um *link* j é um *link* de saída quando existe um fluxo de entrada u_j^l saindo do nó associado a ele.

A Figura 4.2 ilustra os dois tipos de *links* que é possível conectar em um nó e os fluxos associados a eles. A figura define como é representado graficamente cada um dos *links*. Quando a via é de mão dupla os *links* utilizados para representá-la são os mesmos, entretanto, eles são duplicados, define-se um *link* para cada direção.

Passo 2: Identificação dos movimentos não conflitantes

Movimentos são possíveis deslocamentos de veículos numa interseção. Determinados movimentos podem ser conflitantes, ou seja, as suas trajetórias podem ser concorrentes em determinados pontos de um cruzamento.

Figura 4.2: Tipos de *links* que se conectam a um nó

O segundo passo a ser realizado é a identificação dos movimentos em cada *link*. De posse destes movimentos é possível definir conjuntos de movimentos não conflitantes em relação ao nó, onde os elementos destes conjuntos são não conflitantes.

A identificação destes conjuntos é realizada analisando a direção dos fluxos de saída dos *links* de entrada, ou seja, verificando quais são as direções e conversões permitidas em um nó.

É necessário conhecer os conjuntos de movimentos não conflitantes para poder definir os estágios do sistema e a distribuição do fluxo viário. Entretanto, a tarefa de definir estes conjuntos nem sempre é trivial, algumas vezes o engenheiro de tráfego opta pelo uso de regras de prioridade para resolver conflitos difíceis de resolver através dos semáforos.

Passo 3: Definição da matriz de incidência

O terceiro passo é a definição de uma matriz de incidência que indica como é o escoamento do fluxo veicular que atravessa um nó.

Sendo n o número de *links*, define-se uma matriz $n \times n$, onde as linhas da matriz indicam como é a distribuição do fluxo de saída u^o dos *links* de entrada que estão conectados ao nó. As colunas da matriz indicam como é formado o fluxo u^l dos *links* de saída que se conectam ao nó. A equação que define o fluxo u^l é:

$$u_k^l(t) = \sum_{\substack{j \in I \\ j \neq k}} a_{jk} u_j^o(t) \quad (4.1)$$

onde a_{jk} é o elemento da matriz que indica o percentual do fluxo de saída u_j^o do *link* de entrada j que escoou pelo *link* de saída k . O conjunto I contém os índices relativos aos *links* de entrada do nó.

A Figura 4.3 ilustra o exemplo de um sistema composto de uma interseção isolada sinalizada. Além disso, a Figura 4.3(b) indica os *links* identificados no sistema. Pode-se observar que existem dois *links* de entrada (x_1 e y_1) e dois *links* de saída (x_2 e y_2). Aos *links* de entrada x_1 e y_1 estão associados os fluxos de entrada u_{x1}^i e u_{y1}^i e os fluxos de saída u_{x1}^o e u_{y1}^o , respectivamente. Aos *links* de saída x_2 e y_2 estão associados os fluxos u_{x2}^l e u_{y2}^l .

Observando as conversões possíveis na interseção sinalizada (as setas da Figura 4.3(a)) podemos escrever as equações que representam os fluxos u^l (conforme a equação (4.1)) como:

$$u_{x2}^l(t) = a_{12} u_{x1}^o(t) + a_{32} u_{y1}^o(t)$$

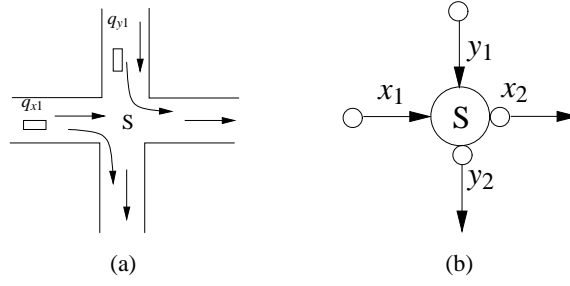


Figura 4.3: Exemplo de uma interseção isolada

$$u_{y2}^l(t) = a_{14} u_{x1}^o(t) + a_{34} u_{y1}^o(t)$$

sendo $a_{jk} \in A$ com $j, k = 1, \dots, 4$. A matriz de incidência A é definida pela matriz da Tabela 4.1.

Ent\Sai	x_1	x_2	y_1	y_2
x_1	0	0.5	0	0.5
x_2	0	0	0	0
y_1	0	0.5	0	0.5
y_2	0	0	0	0

Tabela 4.1: Matriz de incidência A

Analisando a matriz é possível identificar como é a distribuição dos fluxos de saída dos *links* de entrada. Por exemplo, o elemento $a_{12} = 0.5$ indica que 50% do fluxo de saída u_{x1}^o do *link* de entrada x_1 escoam pelo *link* de saída x_2 . Além disso, é possível afirmar, por exemplo, que o fluxo u^l do *link* de saída y_2 é formado pelos fluxos de saída u^o dos *links* de entrada y_1 e x_1 , com os percentuais indicados na matriz.

Quando modelamos apenas uma interseção a distribuição dos fluxos de saída dos *links* de entrada não tem relevância para definir as dinâmicas das filas que podem se formar nestes *links*. Entretanto, quando modelamos sistemas compostos por interseções conectadas é necessário conhecer u^l do *link* de saída que está conectando os nós, pois em relação ao segundo nó este *link* é classificado como *link* de entrada, e u^l define o seu fluxo de entrada. Por fim, no âmbito desta tese utiliza-se apenas matrizes de incidência que não são variantes no tempo, ou seja, a distribuição dos fluxos permanece constante ao longo do tempo.

Passo 4: Determinação dos estágios e fases

O quarto passo a ser realizado é a identificação das fases do sistema. Após identificar os conjuntos de movimentos não conflitantes é necessário atribuir porções do tempo de ciclo de verde a eles, ou seja, definir as fases e os estágios. Quando o semáforo está verde para um determinado estágio significa que uma ou mais fases estão presentes. A porção de tempo de verde alocada para cada estágio é definida pelo *split*.

Cada estágio é definido como um sub-período do ciclo em que um conjunto de movimentos

permitidos está autorizado, ou seja, está com permissão de passagem na interseção. Quando se altera a permissão, muda-se o estágio.

A alocação dos tempos de verde aos estágios é definida de maneira *off-line* quando o controle implementado é de tempo fixo, ou seja, definem-se os estágios e as suas fases *a priori*. Para os casos onde o controle aplicado é atuado a alocação dos tempos semafóricos pode ser *on-line*, onde a duração das fases em cada estágio pode ser realizada dinamicamente, dependendo da demanda de tráfego.

Analisando o exemplo da Figura 4.3 identifica-se dois conjuntos de movimentos não conflitantes relacionados a um nó, e uma configuração possível para o semáforo é composta de dois estágios e duas fases, como ilustrado na Figura 4.4. Se o semáforo está vermelho na direção y e verde em x (50% do tempo de ciclo, o tempo de amarelo é desconsiderado), a fase presente é a fase 1. Já quando o semáforo está verde na direção y e vermelho em x (50% do tempo de ciclo), a fase presente é a fase 2.

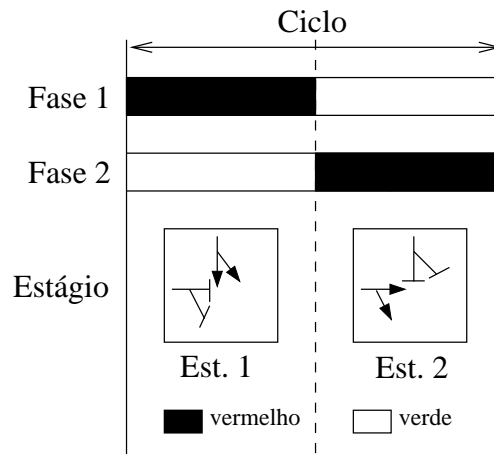


Figura 4.4: Diagrama de estágios e fases para interseção isolada

Passo 5: Formalização das dinâmicas:

As variáveis que serão utilizadas para definir as dinâmicas do sistema são as filas que se formam nos *links* de entrada. Uma fila é o acúmulo de veículos em um *link* de entrada, e ela é formada quando a passagem de veículos em um nó está bloqueada ou quando o fluxo de entrada de veículos u^i é maior que o fluxo de saída u^o .

Quando existem vias de mão dupla, a formação de fila pode ocorrer em ambas direções, e neste caso, para cada direção define-se uma variável distinta.

As equações seguem o modelo definido no trabalho de Lei e Ozguner [2001], onde as filas são modeladas como a variação dos fluxos de entrada e saída de veículos. No trabalho de Lei e Ozguner [2001] assume-se que o tempo de ciclo, fase e *offset* são fixos. A equação apresentada a seguir descreve o comportamento genérico de uma fila em um *link* de entrada j :

$$\dot{q}_j = u_j^i - u_j^o \quad (4.2)$$

onde q_j é o tamanho da fila, u_j^i é o fluxo de entrada e u_j^o é o fluxo de saída no *link*. Para cada estágio do sistema são definidas as equações que modelam as variações das filas.

Passo 6: Definição da lei de controle

Antes de obter o autômato híbrido que modela o sistema é preciso definir a lei de controle que vai ser implementada. O controle pode ser de temporização fixa, onde os tempos de ciclo, *offset* e *split* são definidos de maneira *off-line* e não são alterados com o sistema em funcionamento. Nestes casos o passo 4 é muito importante, pois os estágios e as fases não se alteram.

Outra estratégia de controle que pode ser aplicada é o controle atuado, onde os tempos são definidos dinamicamente, em resposta à demanda de tráfego. Nesta tese apenas a estratégia em tempo fixo será considerada.

Passo 7: Definição do autômato híbrido

Por fim, é necessário definir o autômato híbrido que representa o modelo geral do sistema. Para cada estágio define-se um estado, e dentro deste estado valem as dinâmicas definidas para este estágio quando uma fase está presente. A troca dos estágios é modelada através das guardas e para isto é necessário saber a lei de controle que controla o sistema semafórico. O autômato resultante modela apenas as situações onde ao menos uma fase está presente para cada estágio, ou seja, não se modela a situação de vermelho total, onde nenhum estágio possui permissão de passagem.

4.2.1 Sistemas Compostos por Interseções Conectadas

A proposta de metodologia apresentada tem o intuito de modelar tanto uma interseção isolada como interseções conectadas, onde é necessário modelar a coordenação entre os semáforos. Quando modelamos interseções conectadas devemos considerar interseções separadas por uma distância D e onde existe um *offset* entre os seus semáforos. O tempo que os veículos levam para percorrer a distância entre as interseções é o tempo de deslocamento d .

A modelagem de interseções conectadas considera o sistema como múltiplas interseções isoladas, onde se analisa cada interseção separadamente. Para cada nó identificado são seguidos os sete passos descritos anteriormente, e o acoplamento do sistema se dá através de uma conexão. O acoplamento do sistema implementa a coordenação entre semáforos adjacentes.

Uma conexão liga dois nós subseqüentes, chamados nó-ant e nó-post. Ela é caracterizada por dois *links* distintos, um que se conecta ao nó-ant e outro que se conecta ao nó-post. Conectado ao nó-ant está o *link* de saída *ant*, e associado a ele está o fluxo u_{ant}^l , definido conforme a equação (4.1). Conectado ao nó-post está o *link* de entrada *post*, e associado a ele estão os fluxos de entrada u_{post}^i e u_{post}^o .

A Figura 4.5 ilustra como uma conexão pode ser representada graficamente, respeitando a definição dos *links* de entrada e saída. Quando a via é de mão dupla define-se uma conexão para cada direção da via.

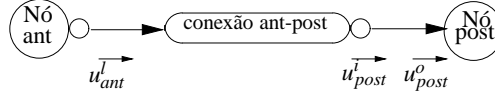


Figura 4.5: Exemplo de conexão em uma interseção isolada

A dinâmica que modela a conexão está relacionada ao *link post*, devido ao fato de ser neste *link* que pode ocorrer formação de fila. A equação que define a variação da fila q_{post} neste *link* é:

$$\dot{q}_{post}(t) = u_{post}^i(t) - u_{post}^o(t).$$

Observando a conexão da Figura 4.5 percebe-se que o fluxo de entrada u_{post}^i é formado pelo fluxo u^l que parte do *link ant* d segundos antes, ou seja, este fluxo alcança o *link post* atrasado d segundos, onde d é o tempo de deslocamento entre os nós. Sendo assim, é possível definir a equação da dinâmica da fila q_{post} em função do fluxo u_{ant}^l , e desta maneira acoplar o sistema. A equação agora é:

$$\dot{q}_{post}(t) = u_{ant}^l(t - d) - u_{post}^o(t).$$

Entretanto, a modelagem de atrasos não é desejada pois deseja-se realizar verificação formal dos sistemas modelados, e os métodos existentes não tratam atrasos. Sendo assim, é necessário utilizar artifícios que tratem os atrasos de maneira implícita.

Nesta tese os atrasos são tratados na definição das guardas que chaveiam os estágios do sistema, e não nas equações que modelam as filas. A idéia é registrar os momentos onde ocorrem chaveamentos de dinâmicas que impliquem na alteração dos fluxos u_{ant}^l , e de posse destes valores calcular qual é o instante que ocorre a mudança de fluxo de entrada em u_{post}^i . A definição das guardas considera os valores registrados, o tempo de deslocamento d e o *offset*, modelando de maneira implícita o atraso das equações das filas.

4.2.2 Aplicação da Metodologia a uma Interseção Isolada

A Figura 4.6(a) apresenta um exemplo de sistema de tráfego, uma interseção isolada, onde as setas indicam os movimentos permitidos nos *links* definidos para o sistema. Estes *links* são mostrados na Figura 4.6(b). Uma matriz de incidência para este sistema está representada na Tabela 4.2.

As variáveis dos sistemas de tráfego urbano que serão analisadas são as filas formadas nos *links* de entrada. Observando a Figura 4.6(a) e 4.6(b) é possível identificar dois *links* de entrada, os *links* x_1 e y_1 . Neste caso, as variáveis que serão modeladas são as filas nestes *links*. Os estágios definidos para o sistema podem ser observados na Figura 4.7. No estágio 1 a fase 2 está presente para os movimentos

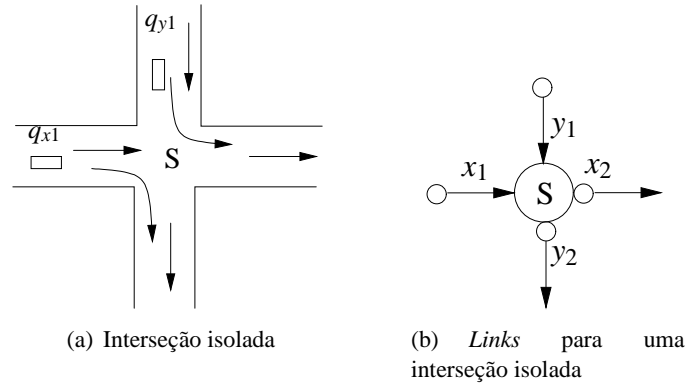


Figura 4.6: Exemplo 1 – interseção isolada

Ent\Sai	x_1	x_2	y_1	y_2
x_1	0	0.7	0	0.3
x_2	0	0	0	0
y_1	0	0.3	0	0.7
y_2	0	0	0	0

Tabela 4.2: Matriz de incidência do semáforo S

do link y_1 e no estágio 2 a fase 1 está presente para os movimentos do link x_1 . No caso da Figura 4.7 o tempo de verde para cada estágio é 1/2 ciclo.

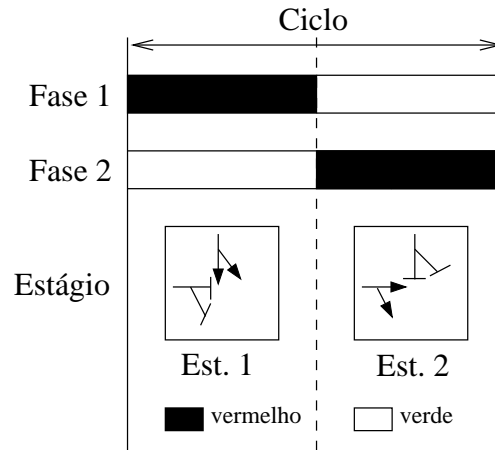


Figura 4.7: Estágios e fases

Sendo q_{x1} e q_{y1} as variáveis contínuas que representam as filas nos links x_1 e y_1 , u^i e u^o os fluxos de entrada e saída de veículos, as dinâmicas das filas para cada estágio estão nas equações a seguir:

$$\text{verde}_{x1} = \begin{cases} \dot{q}_{x1} = u_{x1}^i - u_{x1}^o \\ \dot{q}_{y1} = u_{y1}^i \end{cases} \quad (4.3)$$

$$\text{vermelho}_{x1} = \begin{cases} \dot{q}_{x1} = u_{x1}^i \\ \dot{q}_{y1} = u_{y1}^i - u_{y1}^o \end{cases} \quad (4.4)$$

A equação (4.3) modela as dinâmicas das filas quando o semáforo está verde na direção x (estágio 2). O tamanho da fila q_{x1} varia segundo uma taxa baseada na diferença entre os fluxos de entrada e saída de veículo, e a fila q_{y1} acumula veículos segundo uma taxa que é o seu fluxo de entrada. A equação (4.4) modela as dinâmicas das filas quando o semáforo está verde na direção y (estágio 1), e neste caso, o escoamento de veículos ocorre na fila q_{y1} e o acúmulo de veículos na fila q_{x1} .

A Figura 4.8 mostra o autômato obtido para o exemplo da Figura 4.6(a). A lei de controle definida para o sistema é a de tempo fixo, onde os tempos de verde e vermelho do semáforo são definidos *offline*. A guarda *fim-verde* indica o término da fração de tempo definida para o estágio em que o semáforo está verde na direção x , e, analogamente, *fim-vermelho* indica o término da fração de tempo definida para o estágio em que o semáforo está vermelho na direção x .

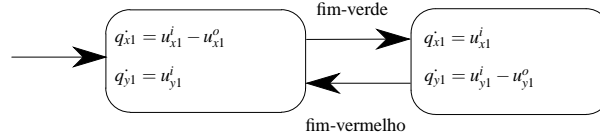


Figura 4.8: Autômato em alto nível para uma interseção isolada

4.3 Modelos Refinados

O autômato obtido na seção 4.2 é uma visão abstrata do sistema modelado como um sistema híbrido. Entretanto, para poder analisar o sistema é importante obter modelos mais detalhados, onde o comportamento do sistema é analisado com maior grau de profundidade. Para definir modelos refinados do autômato híbrido obtido na seção anterior, lembrando que para o caso de múltiplas interseções existe um autômato para cada uma delas, é necessário seguir os seguintes passos:

1. Analisar cada estado do autômato em alto nível (que representam os estágios do sistema) e verificar se ele não poderia ser modelado com mais detalhes. Isto é feito identificando diferentes situações que o sistema pode se encontrar naquele estágio. A característica que se busca identificar é a formação ou não de fila em cada estágio.
2. Para cada nova situação identificada atribuir um estado no autômato híbrido, ou seja, explodir o estado que representa um estágio em outros estados. Sendo assim, é necessário redefinir as dinâmicas do sistema, pelo refinamento da equação (4.2).
3. Para sistemas com múltiplas interseções é necessário avaliar como o tempo de deslocamento entre os semáforos influencia a modelagem das novas situações de comportamento do sistema.
4. Definir como as guardas serão implementadas, principalmente em sistemas com interseções conectadas, de modo a garantir o acoplamento correto do sistema.

Para exemplificar a proposta de refinamento são explorados dois exemplos, um sistema composto de uma interseção isolada e um sistema composto de duas interseções conectadas.

4.3.1 Exemplo 1 – Sistema de uma Interseção Isolada

O autômato da Figura 4.8 foi definido para o sistema da Figura 4.6(a), sendo que a lei de controle aplicada foi a de tempo fixo. O primeiro passo do refinamento é analisar como o sistema se comporta em cada estágio, e é possível identificar duas situações distintas em cada estado: a presença ou não de formação de fila.

Quando existe fila o fluxo de veículos que cruzam o semáforo (nó S) é assumido máximo, ou seja, o fluxo de saída é assumido ser o fluxo de saturação f . Quando não há presença de fila assume-se que o fluxo u^o que parte do semáforo é igual ao que chega (u^i). Como o fluxo de entrada nunca é nulo (considera-se constante e menor que o fluxo de saturação), sempre que um semáforo transita para o sinal verde ele está no modo com fila, devido à presença da fila formada durante o tempo de vermelho do semáforo nesta direção. Após um período de tempo a fila pode esgotar e ele entrar no modo sem fila.

Sendo assim, o autômato encontrado no alto nível pode ser remodelado com quatro estados, onde cada estágio é representado por 2 estados (modo com e sem fila). As novas dinâmicas do sistema quando o semáforo está verde na direção x , ou seja, no estágio 2 da Figura 4.7 são apresentadas a seguir:

$$\text{com fila}_{x1} = \begin{cases} \dot{q}_{x1} = u_{x1}^i - u_{x1}^o = u_{x1}^i - f_{x1} \\ \dot{q}_{y1} = u_{y1}^i - u_{y1}^o = u_{y1}^i - 0 = u_{y1}^i \end{cases} \quad (4.5)$$

$$\text{sem fila}_{x1} = \begin{cases} \dot{q}_{x1} = u_{x1}^i - u_{x1}^o = u_{x1}^i - u_{x1}^i = 0 \\ \dot{q}_{y1} = u_{y1}^i - u_{y1}^o = u_{y1}^i - 0 = u_{y1}^i \end{cases} \quad (4.6)$$

As próximas equações indicam as dinâmicas do sistema quando o semáforo está vermelho na direção x , ou seja, no estágio 1 da Figura 4.7.

$$\text{com fila}_{y1} = \begin{cases} \dot{q}_{x1} = u_{x1}^i - u_{x1}^o = u_{x1}^i - 0 = u_{x1}^i \\ \dot{q}_{y1} = u_{y1}^i - u_{y1}^o = u_{y1}^i - f_{y1} \end{cases} \quad (4.7)$$

$$\text{sem fila}_{y1} = \begin{cases} \dot{q}_{x1} = u_{x1}^i - u_{x1}^o = u_{x1}^i - 0 = u_{x1}^i \\ \dot{q}_{y1} = u_{y1}^i - u_{y1}^o = u_{y1}^i - u_{y1}^i = 0 \end{cases} \quad (4.8)$$

Como o sistema não possui interseções conectadas o passo 3 descrito na seção 4.3 pode ser desconsiderado. A Figura 4.9 ilustra um autômato obtido para um modelo refinado de uma interseção isolada. Neste autômato cada estágio é representado por dois estados, e é necessário definir guardas que possibilitem a transição entre estados que representam o mesmo estágio, devido ao esgotamento da fila em determinada direção.

Para implementar as guardas é possível utilizar variáveis definidas como relógios para contar o tempo de ciclo, e no caso do autômato da Figura 4.9 é necessário apenas uma, a variável denominada

r . Esta variável, r , assume valores reais dentro do conjunto $[0, T]$, ou seja, a duração do tempo de ciclo T . Sempre que $r = T$ ela é reinicializada e assume valor $r = 0$, indicando o início de um novo ciclo. Para que o modelo esteja correto é necessário adicionar a cada estado do autômato da Figura 4.9 a dinâmica contínua que modela a variável r , ou seja, em cada estado deve ser adicionado a dinâmica $\dot{r} = 1$.

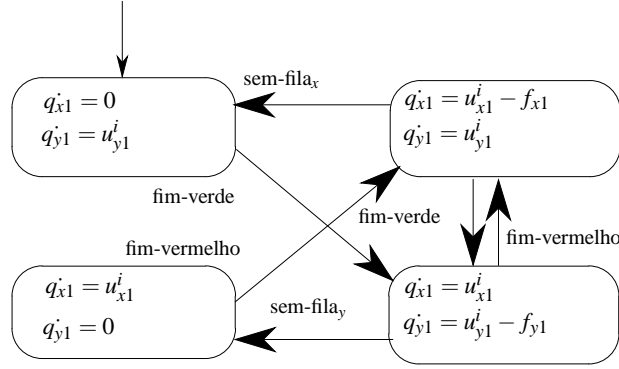


Figura 4.9: Autômato do modelo refinado para uma interseção isolada

A Tabela 4.3 mostra como as guardas do autômato da Figura 4.9 podem ser representadas em função de parâmetros do sistema, onde T é o tempo de ciclo, c o *split*, q_{x1} e q_{y1} as filas e r um relógio que conta o tempo de ciclo. Um novo ciclo sempre inicia com o semáforo em verde na direção x , ou seja, no intervalo $[0, cT]$ o semáforo está em verde na direção x (estágio 2) e no intervalo $(cT, T]$ ele está vermelho nesta direção (estágio 1).

Etiqueta	Guarda
fim-vermelho	$r = T ; r := 0$
fim-verde	$r = cT$
sem-fila-x	$q_{x1} = 0$
sem-fila-y	$q_{y1} = 0$

Tabela 4.3: Guardas do autômato para uma interseção isolada

O autômato resultante pode ser visualizado na Figura 4.10, onde cada caixa pontilhada corresponde a um estado do modelo de alto nível da Figura 4.8.

4.3.2 Exemplo 2 – Sistema com Duas Interseções Conectadas

Suponha o sistema da Figura 4.11, onde existem duas interseções separadas por uma distância D , e o tempo para percorrer esta distância é o tempo de deslocamento d . Os semáforos existentes nas interseções são sincronizados pelo *offset*, que é a defasagem entre os instantes de início do tempo de verde. Os *links* e a conexão do sistema estão representados na Figura 4.12. O modelo em alto nível é composto por dois autômatos, como mostra a Figura 4.13. Neste exemplo não são permitidas conversões, como pode ser observado nos estágios propostos para as interseções na Figura 4.14 (os estágios são iguais para as duas interseções).

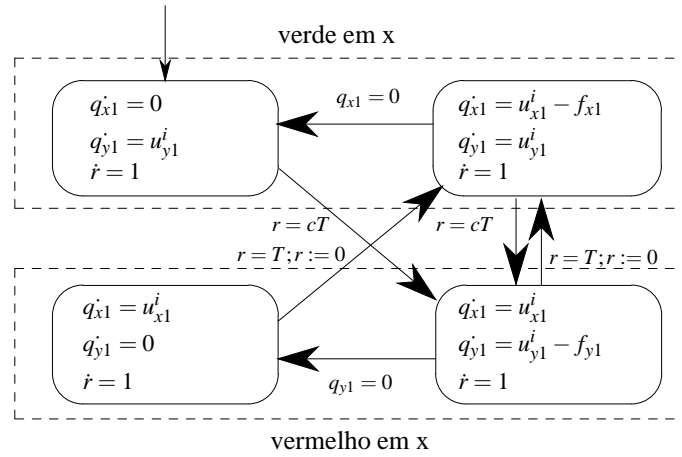


Figura 4.10: Autômato final para o modelo refinado (uma interseção isolada)

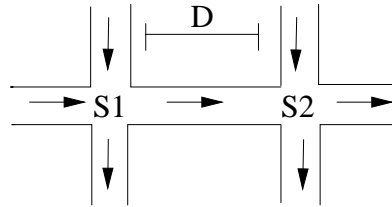


Figura 4.11: Sistema com duas interseções conectadas

O refinamento deve ser efetuado para cada semáforo separadamente, sendo o acoplamento definido nas novas guardas que serão implementadas. Para o autômato que representa o primeiro nó (S1) o procedimento é idêntico ao descrito no exemplo anterior; as diferenças aparecem no modelo do segundo nó, onde o tempo de deslocamento entre os semáforos influencia o chaveamento das dinâmicas. As matrizes de incidência para o primeiro e o segundo nó estão, respectivamente, nas Tabela 4.4 e 4.5.

Ent\Sai	x_1	x_2	y_1	y_2
x_1	0	1	0	0
x_2	0	0	0	0
y_1	0	0	0	1
y_2	0	0	0	0

Tabela 4.4: Matriz de incidência do semáforo S1

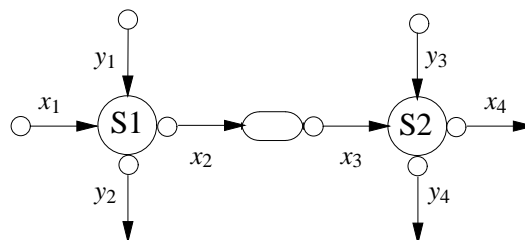


Figura 4.12: Links para o sistema com duas interseções

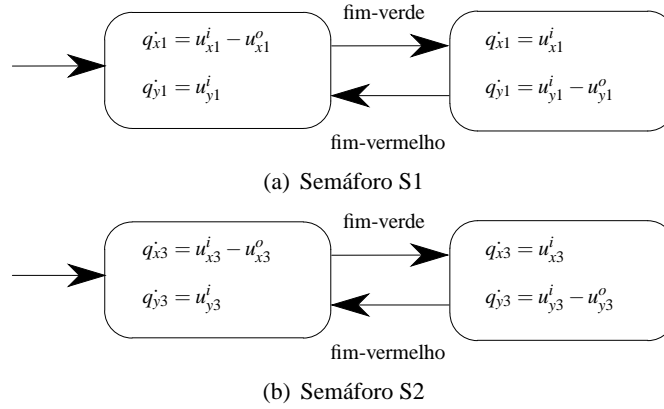


Figura 4.13: Modelos em alto nível para S1 e S2

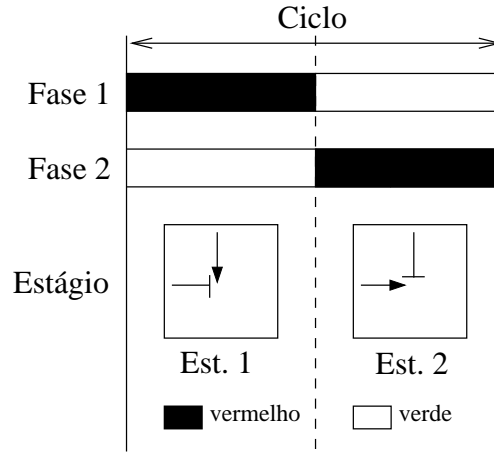


Figura 4.14: Estágios e fases – exemplo 2

Ent\Sai	x_3	x_4	y_3	y_4
x_3	0	1	0	0
x_4	0	0	0	0
y_3	0	0	0	1
y_4	0	0	0	0

Tabela 4.5: Matriz de incidência do semáforo S2

O primeiro passo do refinamento do segundo autômato é identificar diferentes comportamentos das filas em cada estágio, ou seja, em cada estado. Quando o sistema está no estágio em que o semáforo S2 está verde na direção x (estágio 2) é possível identificar novamente duas situações: com e sem fila. Além disso é possível identificar três situações de chegada de veículos no link x_3 , pois eles chegam com um fluxo $u_{x3}^i(t) = u_{x2}^l(t - d)$ que pode ser f_{x1} , u_{x1}^i ou 0 (sendo $u_{x1}^i < f_{x1}$ e ambos constantes).

Quando está vermelho na direção x do semáforo S2 (estágio 1) a formação de fila no link x_3 também depende dos fluxos de veículos que partem do primeiro nó. Considerando estas novas suposições é possível redefinir as dinâmicas do sistema.

As variáveis contínuas do semáforo S2 são as filas q_{x3} e q_{y3} , filas formadas nos *links* x_3 e y_3 , respectivamente, e seus modelos matemáticos quando está verde na direção x (semáforo S2 no estágio 2) estão descritos nas equações a seguir:

$$\text{com fila}_{x3} = \begin{cases} \dot{q}_{x3} = -f_{x3} & \text{caso } u_{x3}^i = 0 \\ \dot{q}_{x3} = u_{x1}^i - f_{x3} & \text{caso } u_{x3}^i = u_{x1}^i \\ \dot{q}_{x3} = f_{x1} - f_{x3} & \text{caso } u_{x3}^i = f_{x1} \\ \dot{q}_{y3} = u_{y3}^i & \end{cases} \quad (4.9)$$

$$\text{sem fila}_{x3} = \begin{cases} \dot{q}_{x3} = 0 \\ \dot{q}_{y3} = u_{y3}^i \end{cases} \quad (4.10)$$

Este estágio agora precisa de 4 estados para ser modelado através de um autômato híbrido.

As equações que representam as dinâmicas na direção x quando o semáforo S2 está com o sinal vermelho nesta direção (estágio 1) são:

$$\text{com fila}_{y3} = \begin{cases} \dot{q}_{x3} = 0 & \text{caso } u_{x3}^i = 0 \\ \dot{q}_{x3} = u_{x1}^i & \text{caso } u_{x3}^i = u_{x1}^i \\ \dot{q}_{x3} = f_{x1} & \text{caso } u_{x3}^i = f_{x1} \\ \dot{q}_{y3} = u_{y3}^i - f_{y3} & \end{cases} \quad (4.11)$$

$$\text{sem fila}_{y3} = \begin{cases} \dot{q}_{x3} = 0 & \text{caso } u_{x3}^i = 0 \\ \dot{q}_{x3} = u_{x1}^i & \text{caso } u_{x3}^i = u_{x1}^i \\ \dot{q}_{x3} = f_{x1} & \text{caso } u_{x3}^i = f_{x1} \\ \dot{q}_{y3} = u_{y3}^i - u_{y3}^i = 0 & \end{cases} \quad (4.12)$$

Para modelar este estágio são necessários 6 estados no autômato híbrido pois é preciso considerar as diferentes chegadas em x quando o semáforo está vermelho nesta direção.

Após definir as novas dinâmicas para o sistema é necessário analisar como o tempo de deslocamento d será considerado na modelagem. A proposta é traduzi-lo nas guardas do autômato híbrido. Assim, como no exemplo para uma interseção isolada, aqui também se utilizam variáveis definidas como relógios. Através destas variáveis é possível saber em que momento aconteceram as trocas de dinâmicas no primeiro semáforo, e assumindo os tempos de deslocamento constantes é possível prever quando ocorrem trocas de dinâmicas no segundo semáforo.

A Figura 4.15 ilustra esta idéia, onde s é o *offset* entre os semáforos, c o *split*, T o tempo de ciclo e d o tempo de deslocamento entre os nós. São definidas duas variáveis como relógios, r_1 e r_2 . O relógio r_1 controla o tempo de ciclo e r_2 registra o momento da troca de dinâmica no semáforo S1. O ciclo sempre começa com o semáforo verde para o nó S1, e é possível perceber na situação descrita na Figura 4.15 que os veículos que partem do semáforo S1 no instante $r_1 = 0$ alcançam o segundo semáforo d segundos depois (no caso $r_1 = s$), exatamente quando o semáforo S2 ficou verde.

As variáveis são zeradas a cada novo ciclo. Com estas informações é possível modelar guardas que controlam os tempos de verde e vermelho nos dois nós (S1 e S2) e guardas que acoplem os dois semáforos.

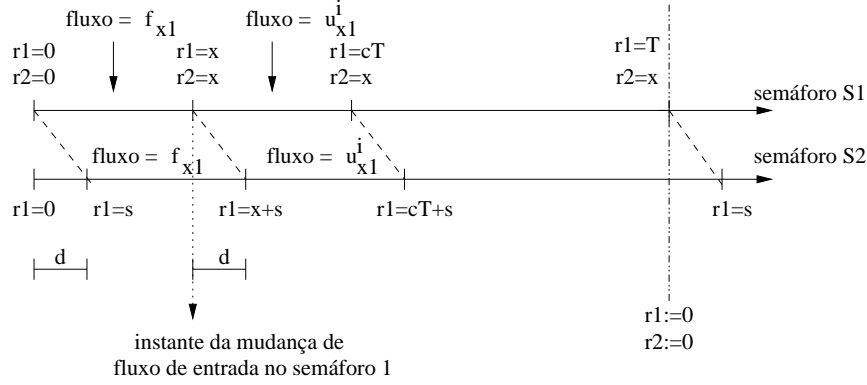


Figura 4.15: Relógios para modelar guardas

A Figura 4.16 ilustra o autômato refinado obtido para o segundo semáforo. As guardas definidas para este autômato estão na Tabela 4.6, lembrando que s é *offset*, T o tempo de ciclo, d o tempo de deslocamento, c o *split*, r_1 e r_2 as variáveis que representam relógios, e q_{x3} e q_{y3} as variáveis de estado que representam as filas do nó S2.

Etiqueta	Guarda
a	$q_{x3} = 0$
b	$q_{y3} = 0$
c	$r_1 = s$
d	$r_1 \geq cT + s$
e	$r_1 - d = r_2$
f	$r_1 \geq d$
g	$r_1 \geq cT + d$
h	$r_1 = s \wedge q_{x3} = 0$

Tabela 4.6: Guardas do autômato do semáforo S2

As guardas a e b são habilitadas quando as filas q_{x3} e q_{y3} são nulas. O término do tempo de verde ou vermelho é sinalizado através das guardas d e c , respectivamente, sendo que é utilizado um único relógio para controlar os tempos de ciclo dos dois semáforos, e isto é possível porque utiliza-se o *offset* s entre os semáforos para sincronizá-los.

A guarda e tem o intuito de sinalizar a mudança no fluxo de entrada de veículos no semáforo S2, considerando o instante que os veículos deixaram o semáforo S1. Para este teste utiliza-se a variável r_2 , que indica quando o semáforo S1 deixou de atuar no modo com fila e onde o fluxo era f_{x1} . O teste consiste em verificar no semáforo S2 se o instante em que o veículo partiu do semáforo S1 ($r_1 - d$) ocorreu antes do instante armazenado na variável r_2 , e isto pode ser visualizado na Figura 4.15. Se o veículo partiu antes do instante r_2 a guarda não é habilitada, caso contrário, transita para um estado

dos relógios, pode ser encontrado na Figura 4.18.

É possível perceber no novo autômato a inclusão da dinâmica da variável r_2 , que guarda o tempo que o semáforo S1 está descarregando veículos na direção x no modo com fila. Esta variável assume valores reais dentro do conjunto $[0, cT]$ e é reinicializada no início de cada ciclo (quando a transição associada à guarda *fim-vermelho* ocorre), sendo que a dinâmica neste estado é $\dot{r}_2 = 1$. Quando o sistema transita para outros estados a dinâmica é alterada para $\dot{r}_2 = 0$, armazenando em r_2 o instante da transição de estado. Este instante representa a parcela de tempo que o sistema ficou com o semáforo no sinal verde e descarregando fila na direção x .

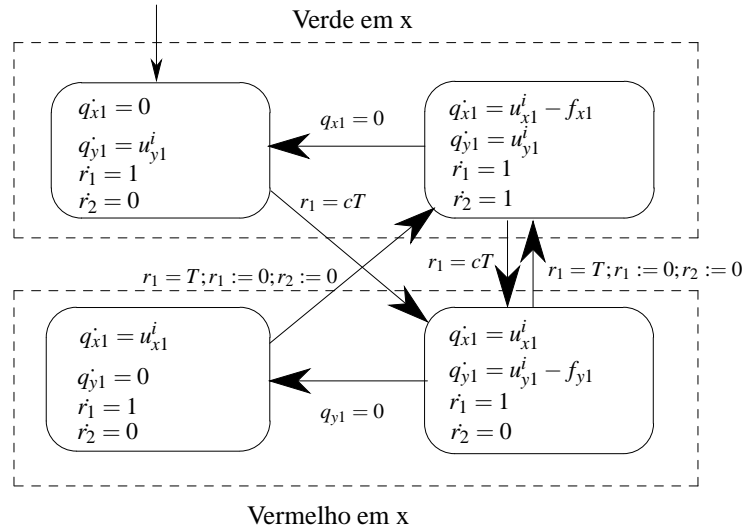


Figura 4.18: Autômato do semáforo S1 considerando as dinâmicas dos relógios

4.4 Implementação

A modelagem proposta nos exemplos desenvolvidos é implementada com a ajuda de uma ferramenta computacional, o *CheckMate* [Silva et al., 2000]. O *CheckMate* é uma ferramenta baseada no *MATLAB* para modelagem, simulação e verificação formal de sistemas híbridos.

4.4.1 Ferramenta Utilizada – *CheckMate*

A ferramenta *CheckMate* é implementada no *MATLAB*, utilizando a interface gráfica *Simulink*. O *CheckMate* é composto por três blocos principais: *switched continuous system block*, *polyhedral threshold block* e *finite state machine block*, que podem ser observados na Figura 4.19. Estes três blocos são utilizados juntamente com os blocos padrões do *Simulink* para construir modelos de sistemas híbridos que podem ser simulados e verificados [Chutinan e Krogh, 2003].

O *switched continuous system block* (SCSB) define um sistema dinâmico contínuo cujas dinâmicas dependem de uma entrada de valor discreto. Um SCSB possui uma entrada discreta u e uma saída x , que é um vetor de estado contínuo para as dinâmicas do bloco. A dinâmica contínua é selecionada

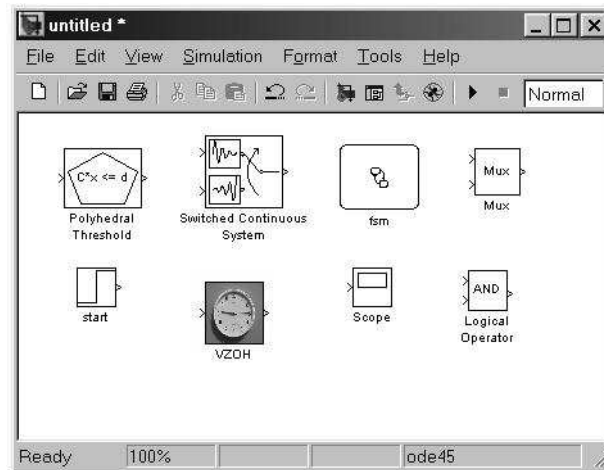


Figura 4.19: Blocos disponíveis no *CheckMate*

pelo valor u de acordo com $\dot{x} = f_u(x)$. A entrada discreta u no SCSB só pode vir da máquina de estado finito. Os tipos de equações diferenciais ordinárias que podem ser especificados para cada valor discreto da entrada u são: não-linear, onde $\dot{x} = f(x)$ para uma função não-linear contínua arbitrária f ; linear, onde $\dot{x} = Ax + b$ para uma matriz constante A e um vetor b ; e constante, onde $\dot{x} = c$ para um vetor constante c . Para cada caso o *CheckMate* utiliza um procedimento de aproximação *flow-pipe*, que é otimizado para o tipo especificado de dinâmica, sendo que a representação *flow-pipe* é exata no caso de dinâmicas constantes.

O *Polyhedral Threshold Block* (PTHB) define um poliedro convexo parametrizado por um par matrix-vetor (C, d) . A entrada é um vetor estado contínuo x e a saída é um sinal booleano indicando se x está ou não dentro do poliedro convexo definido por $Cx \leq d$. A entrada deve ser construída à partir das saídas dos SCSBs.

O bloco *finite-state machine block* (FSMB) é implementado pelo *Stateflow* com as seguintes restrições:

- cada entrada evento deve ser uma função lógica das saídas dos PTHBs;
- cada dado de entrada deve ser uma função lógica das saídas dos PTHBs ou FSMB;
- somente são permitidos sinais de saída com valor discreto;
- o diagrama *Stateflow* não deve conter hierarquias e cada estado deve atribuir um valor único para o dado de saída na ação de entrada. Nenhuma outra ação é permitida em qualquer estado ou transição rotulada.

Além dos blocos desenvolvidos especificamente para o *CheckMate* existem os blocos-padrão do *Simulink*. Para simulação qualquer bloco pode ser utilizado, entretanto para verificação, além dos blocos específicos, podem ser utilizados apenas os operadores lógicos e sinais MUX/DEMUX.

O procedimento de verificação do *CheckMate* é baseado na teoria geral de autômato híbrido com semântica de sistemas de transição. Para aplicar esta teoria o *CheckMate* converte modelos *Simulink-Stateflow* em uma classe de autômatos híbridos chamada *polyhedral-invariant hybrid automata* (PIHA). O PIHA difere de um autômato híbrido geral nos seguintes aspectos:

1. Não existe nenhum mapeamento de *reset* associado às transições discretas, o que significa que não existem descontinuidades nas trajetórias no estado contínuo.
2. Os invariantes são definidos por inequações lineares.
3. As guardas são faces dos invariantes, o que significa que uma transição no espaço discreto ocorre imediatamente quando a trajetória no estado contínuo atinge uma guarda.

Assim como os sistemas híbridos em geral, o sistema de transição do PIHA tem um espaço de estados infinito. Para aplicar técnicas do padrão *model checking* uma aproximação conservativa de estado finito é construída, resultando no chamado sistema de transição quociente aproximativo (AQTS). Se a verificação é inconclusiva, o AQTS pode ser refinado e o procedimento de verificação pode ser executado novamente [Chutinan e Krogh, 2001]. O principal obstáculo em relação à realização do AQTS para o sistema híbrido é a falta de métodos efetivos para computar os *flow-pipes*, ou seja, o conjunto de trajetórias contínuas partindo de um conjunto de estados iniciais. Em Chutinan e Krogh [1998] é proposto um procedimento para computar aproximações poliedrais conservativas para *flow-pipes* de sistemas dinâmicos contínuos.

O procedimento de verificação no *CheckMate* inicia com a conversão do modelo *Simulink* em um PIHA equivalente. A análise do PIHA é limitada pela região poliedral definida pelo usuário chamada região de análise do espaço de estados contínuo. Um AQTS de estado finito é construído para verificar propriedades nas trajetórias de estado do PIHA. Os estados discretos no AQTS são definidos por uma partição inicial da superfície de chaveamento (isto é, os limites dos poliedros definidos nos *threshold blocks*), construídos de acordo com os parâmetros especificados pelo usuário. Transições entre estados no AQTS são computados usando aproximação por *flow-pipes*. O AQTS é então verificado novamente para uma dada especificação usando alguma técnica do padrão *model checking* para sistemas de transição de estados finitos.

O *CheckMate* permite modelar especificações escritas em lógica ACTL. Fórmulas ACTL consistem de operadores temporais e quantificadores de caminho. Se a verificação falha devido à discretização grosseira do AQTS, a partição para o AQTS é refinada. O processo pode ser repetido até que o AQTS satisfaça a especificação ou que o usuário termine a verificação.

4.4.2 Simulação – Resultados Obtidos

O modelo proposto para o sistema composto de duas interseções (Figura 4.11) foi implementado no *CheckMate*. O sistema foi definido como um modelo único composto de 6 variáveis de estados, ou seja, apenas um *switched continuous system block* (SCSB) foi definido para o sistema

$x = [r_1 \ q_{x1} \ q_{y1} \ r_2 \ q_{x3} \ q_{y3}]$. Apesar de utilizar apenas um SCSB, ou seja, definir as variáveis dos semáforos como um sistema global, foram utilizadas duas máquinas de estado, e para cada qual foram definidas guardas. Cada guarda é modelada em um PTHB. A Figura 4.20 apresenta o diagrama de blocos definido no *CheckMate* para executar as simulações.

O primeiro semáforo foi projetado em um FSMB com quatro estados, como mostra a Figura 4.21, e foram definidos 4 PTHBs no modelo da Figura 4.20 para representar as guardas. O FSMB do segundo semáforo foi projetado com 10 estados (Figura 4.22), e 8 PTHBs foram definidos no modelo da Figura 4.20 para implementar as guardas. Os autômatos das Figuras 4.21 e 4.18 são similares, a única diferença é o estado inicial. O mesmo ocorre com os autômatos das Figuras 4.22 e 4.16. Como as dinâmicas foram definidas em apenas um SCSB é necessário definir uma expressão que represente um estado equivalente aos estados dos 2 FSMB. A expressão proposta é $u = 10(u_1 - 1) + u_2$. A dinâmica proposta para o sistema $x = [r_1 \ q_{x1} \ q_{y1} \ r_2 \ q_{x3} \ q_{y3}]$ no estado $u = 4$, por exemplo, representa a dinâmica proposta para o sistema com os semáforos S1 e S2 nos estados $u_1 = 1$ e $u_2 = 4$, respectivamente.

As dinâmicas e as guardas são definidas em programas armazenados em arquivos com extensão .m, compatíveis com o *MATLAB* e específicos para o *CheckMate*. Nestes arquivos também são informados todos os parâmetros relacionados ao sistema, como o tempo de deslocamento, *offset*, tempo de ciclo, valores de fluxo, entre outros.

Foram realizadas diversas simulações para analisar o comportamento do sistema. Alguns resultados obtidos são apresentados a seguir.

Simulação 1:

A simulação que é apresentada aqui considerou que o tempo de deslocamento e o *offset* são diferentes, $s < d$, ou seja, nem todos os carros que partem do semáforo S1 com o sinal verde alcançarão o semáforo S2 com o sinal verde no mesmo ciclo. Os dados da simulação são:

- tempo de ciclo $T = 60$ (segundos);
- *split* $c = 0.5$ (metade do tempo sinal verde e metade vermelho);
- tempo de deslocamento $d = 20$ (segundos);
- *offset* $s = 15$ (segundos);
- fluxo de saturação $f_{x1} = 1800/3600$ (veículos/segundo);
- fluxo de saturação $f_{y1} = 1800/3600$ (veículos/segundo);
- fluxo de saturação $f_{x3} = 1800/3600$ (veículos/segundo);
- fluxo de saturação $f_{y3} = 1800/3600$ (veículos/segundo);

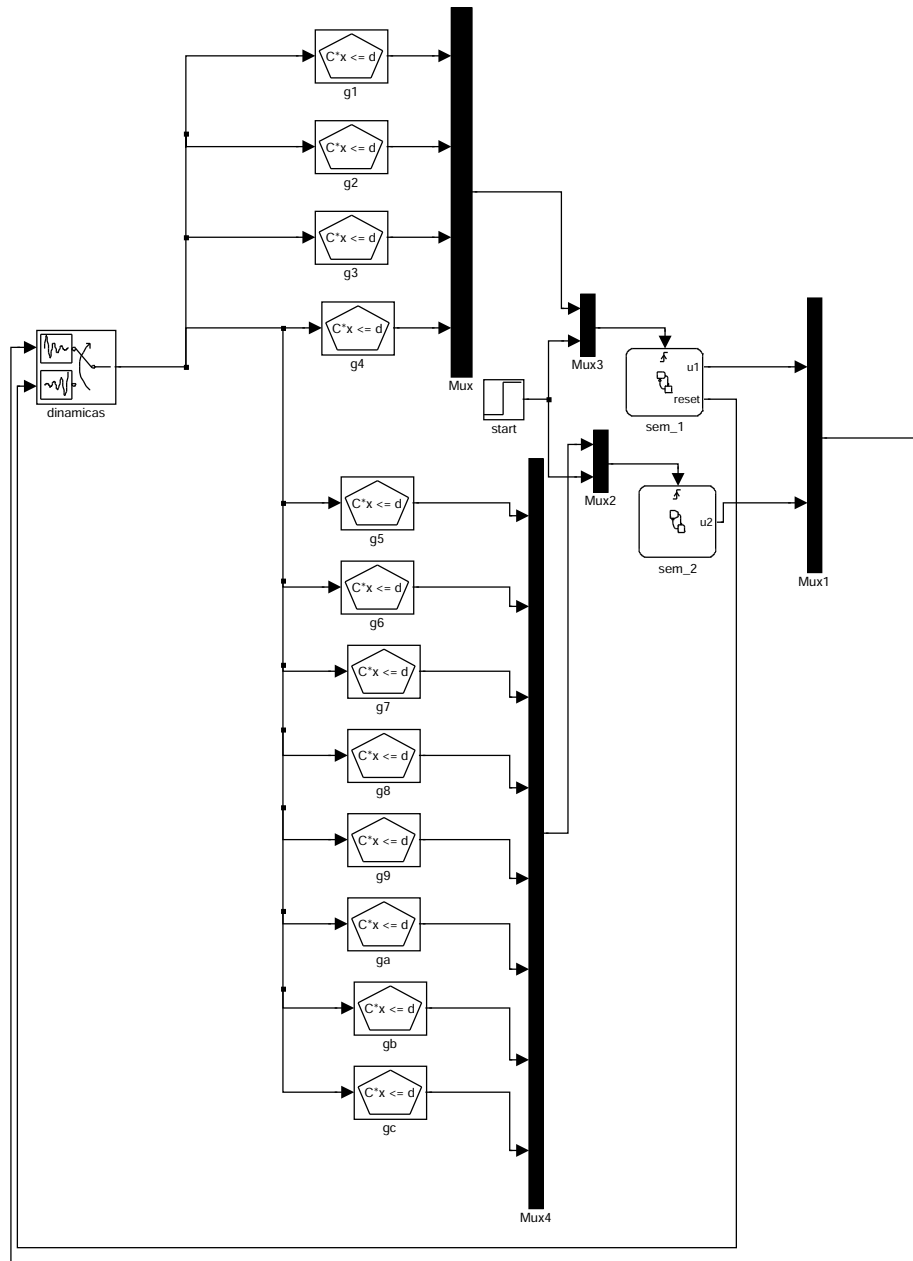


Figura 4.20: Modelo desenvolvido no *CheckMate*

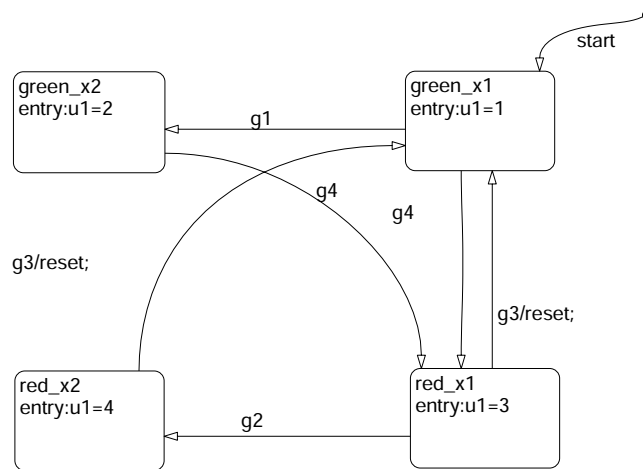


Figura 4.21: FSMB que modela semáforo S1

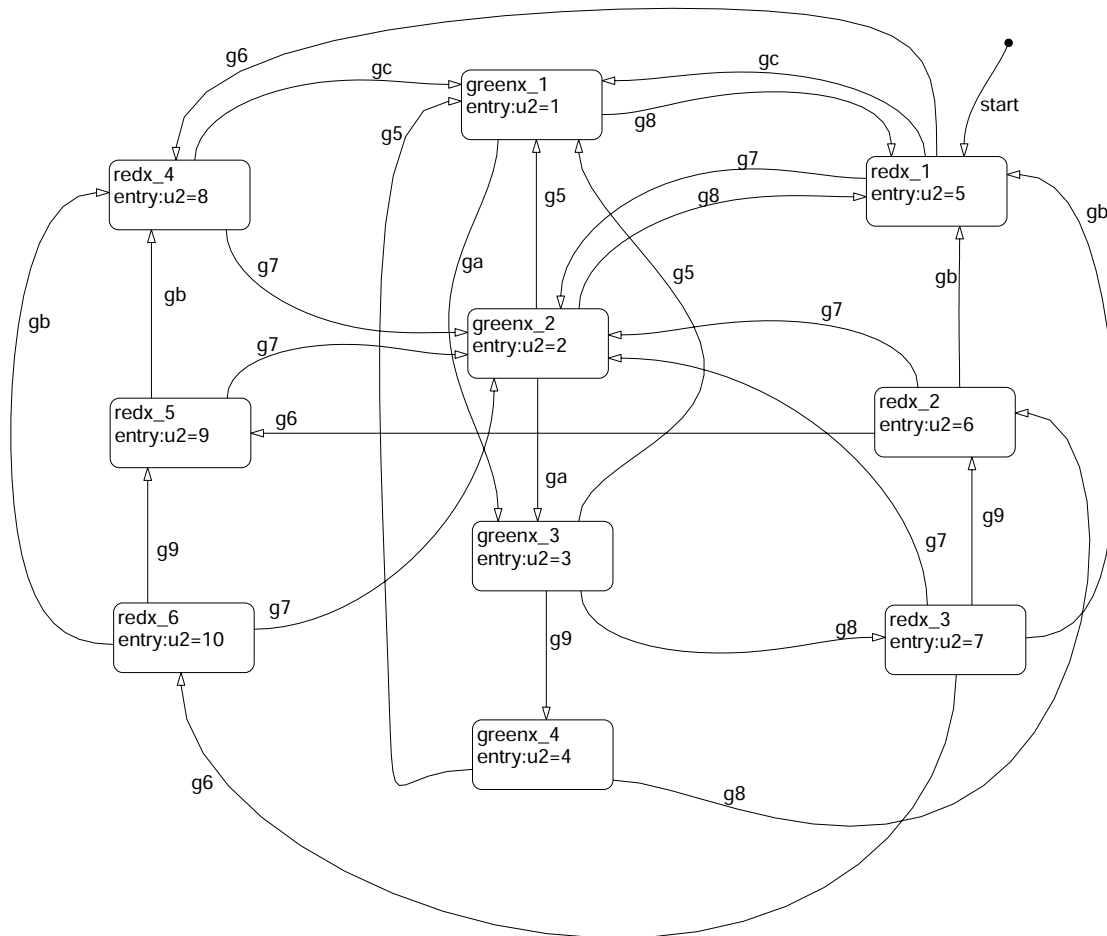


Figura 4.22: FSMB que modela semáforo S2

- fluxo de entrada de veículos $u_{x1} = u_{y1} = 1500/3600$ (veículos/segundo);
- fluxo de entrada de veículos $u_{y3} = 900/3600$ (veículos/segundo);
- condições iniciais: $q_{x1} = 2$, $q_{y1} = 0$, $q_{x3} = 4$ e $q_{y3} = 2$, além disso o semáforo S1 está com o sinal verde e o S2 com o sinal vermelho na direção x . Os valores iniciais definem os estados iniciais dos autômatos de S1 e S2.
- tempo de simulação $t = 300$ (segundos).

A evolução das filas em x e y no semáforo S1 está na Figura 4.23. É possível perceber que o semáforo atua sempre no modo com fila em regime permanente, pois as filas não se esgotam em nenhuma direção. Isto acontece porque a taxa de ocupação da via, em ambas direções, é muito alta. O grau de saturação, que define uma relação entre o número de veículos que chegam no cruzamento durante um ciclo e o número de veículos que podem ser atendidos, tem valor maior que 1, e quando isso acontece significa que chegam mais veículos do que podem ser atendidos (ver Apêndice A).

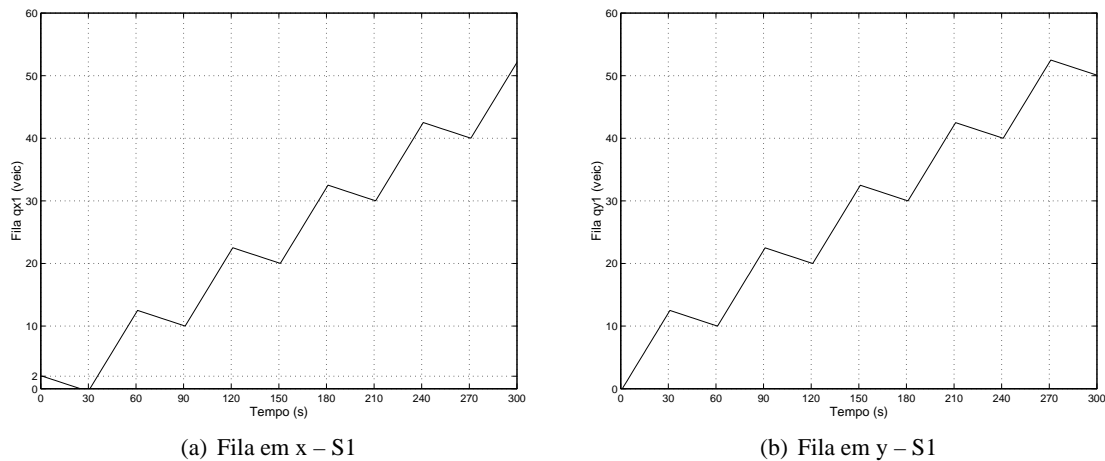


Figura 4.23: Comportamento das filas no semáforo S1 – Simulação 1

A evolução das filas em x e y no semáforo S2 estão na Figura 4.24. Na direção y o semáforo atua no modo com fila e se a fila esgota passa a atuar no modo sem formação de fila (isto ocorre apenas no primeiro ciclo). Em regime permanente a fila q_{y3} se esgota no instante da troca semafórica, enquanto a fila q_{x3} não se esgota devido ao seu estado inicial, que extrapola a capacidade da via. A evolução do sistema nos estados dos autômatos híbridos das Figuras 4.21 e 4.22 pode ser observado no gráfico da Figura 4.25.

O gráfico mostra que, em regime permanente, o autômato de S1 oscila entre os estados $u_1 = 1$ e $u_1 = 3$ da Figura 4.21, o que confirma que o sistema está sempre com formação de fila em ambos os sentidos da via. O autômato da Figura 4.22 percorre a seguinte sequência de estados em regime permanente ($u_2 = 2, u_2 = 3, u_2 = 7, u_2 = 5, u_2 = 2, u_2 = 3, u_2 = 7, \dots$). Na direção x inicialmente existe uma descarga de fila (estado $u_2 = 2$) e quando os veículos que partem de S1 chegam em S2 o sistema atua sempre no modo com fila, onde o fluxo de chegada é sempre igual ao fluxo de saída, no caso o fluxo de saturação f_{x1} . Isto é comprovado com a visita ao estado $u_2 = 7$, onde o semáforo está

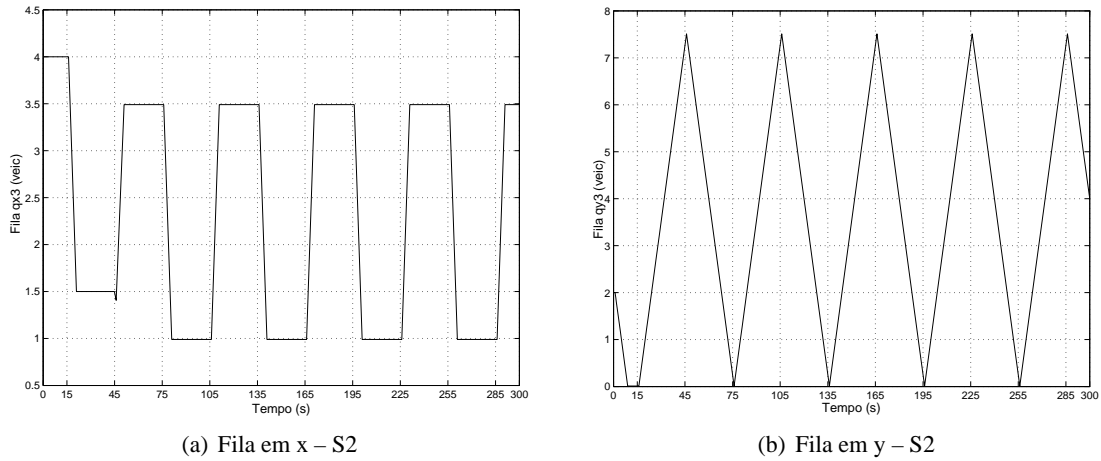


Figura 4.24: Comportamento das filas no semáforo S2 – Simulação 1

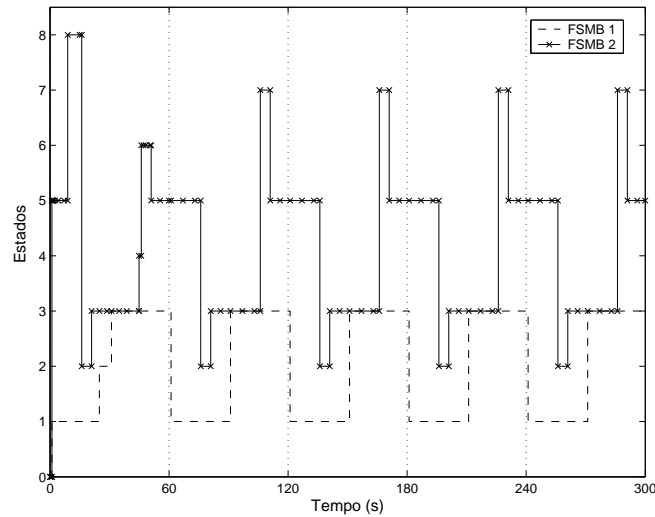


Figura 4.25: Evolução dos estados nos FSMBs – Simulação 1

vermelho em x e a sua fila cresce segundo a taxa f_{x3} . Quando a chegada de veículos cessa o sistema transita para o estado $u_2 = 5$.

O comportamento dos relógios r_1 e r_2 está na Figura 4.26. O relógio r_1 é reinicializado em 0 a cada novo ciclo de S1, e a variável r_2 está sempre no seu valor máximo em regime permanente. Este resultado é o esperado pois o semáforo S1 atua sempre no modo com fila, logo não ocorre mudança de dinâmica enquanto o sinal está verde. Isto comprova o fato do autômato de S2 não entrar no estado $u_2 = 6$.

O regime transitório para esta simulação dura um ciclo de cada semáforo. Através dos gráficos é possível perceber que durante o primeiro ciclo de S1 a fila q_{x1} chega a ser nula durante o sinal verde. O instante que isto ocorre fica registrado no relógio r_2 , e pode ser confirmado no gráfico da Figura 4.26. Além disso a evolução dos estados do semáforo S1 também comprova este fato, pois fica registrado que o estado $u_1 = 2$ foi visitado no primeiro ciclo. A visita aos estados $u_2 = 4$ e $u_2 = 6$ se devem justamente ao fato da fila q_{x1} ser nula por alguns instantes no primeiro ciclo, onde então o fluxo de chegada de veículos em q_{x3} é u_{x1} .

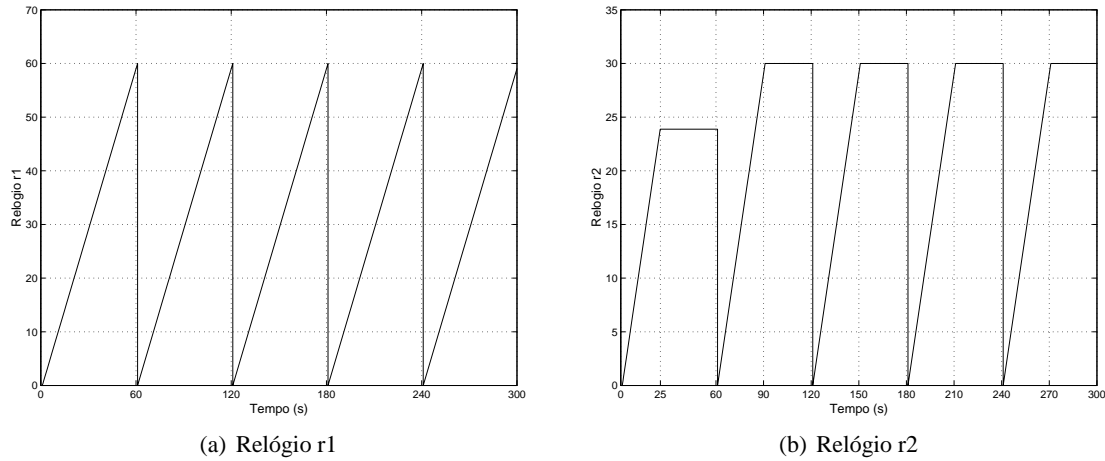


Figura 4.26: Comportamento dos relógios – Simulação 1

Simulação 2:

A simulação apresentada anteriormente mostra que quando a taxa de ocupação da via é muito alta no semáforo S1 não é possível esgotar a fila nos *links* q_{x1} e q_{y1} . O objetivo desta simulação é ver o comportamento do sistema quando as taxas de ocupação nestes *links* não ultrapassam a capacidade da via.

- tempo de ciclo $T = 60$ (segundos);
- *split* $c = 0.5$ (metade do tempo sinal verde e metade vermelho);
- tempo de deslocamento $d = 20$ (segundos);
- *offset* $s = 15$ (segundos);
- fluxo de saturação $f_{x1} = 1800/3600$ (veículos/segundo);
- fluxo de saturação $f_{y1} = 1800/3600$ (veículos/segundo);
- fluxo de saturação $f_{x3} = 1800/3600$ (veículos/segundo);
- fluxo de saturação $f_{y3} = 1800/3600$ (veículos/segundo);
- fluxo de entrada de veículos $u_{x1} = u_{y1} = 900/3600$ (veículos/segundo);
- fluxo de entrada de veículos $u_{y3} = 900/3600$ (veículos/segundo);
- condições iniciais: $q_{x1} = 2$, $q_{y1} = 0$, $q_{x3} = 9$ e $q_{y3} = 2$, além disso o semáforo S1 está com o sinal verde e o S2 com o sinal vermelho na direção x ;
- tempo de simulação $t = 300$ (segundos).

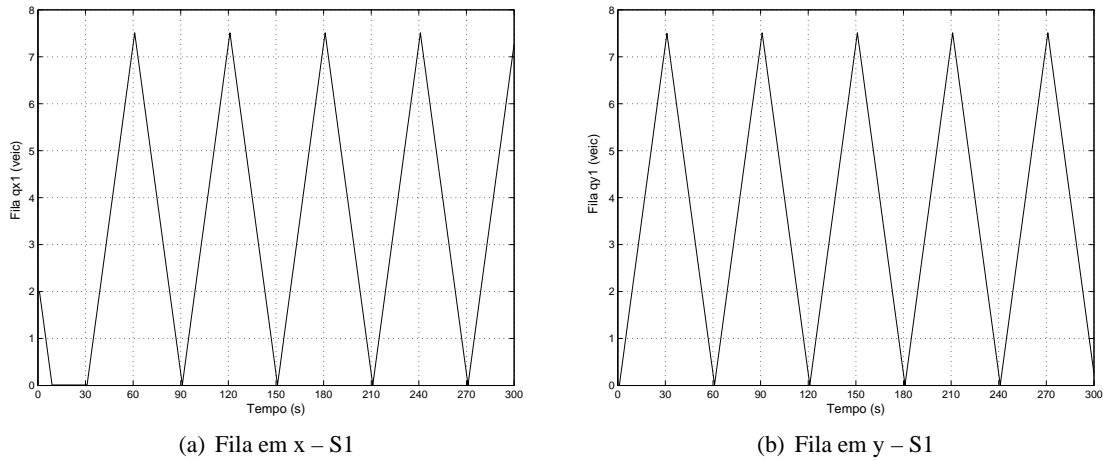


Figura 4.27: Comportamento das filas no semáforo S1 – Simulação 2

A evolução das filas em x e y no semáforo S1 está na Figura 4.27. É possível perceber que nesta simulação as filas se tornam nulas em ambas as direções. Com exceção do primeiro ciclo, as filas se esgotam no instante da troca semafórica.

A evolução das filas em x e y no semáforo S2 estão na Figura 4.28. Em regime permanente o comportamento das filas é igual ao apresentado na simulação anterior. As diferenças aparecem no regime transitório. A evolução do sistema nos estados dos autômatos híbridos das Figuras 4.21 e 4.22 pode ser observado no gráfico da Figura 4.29.

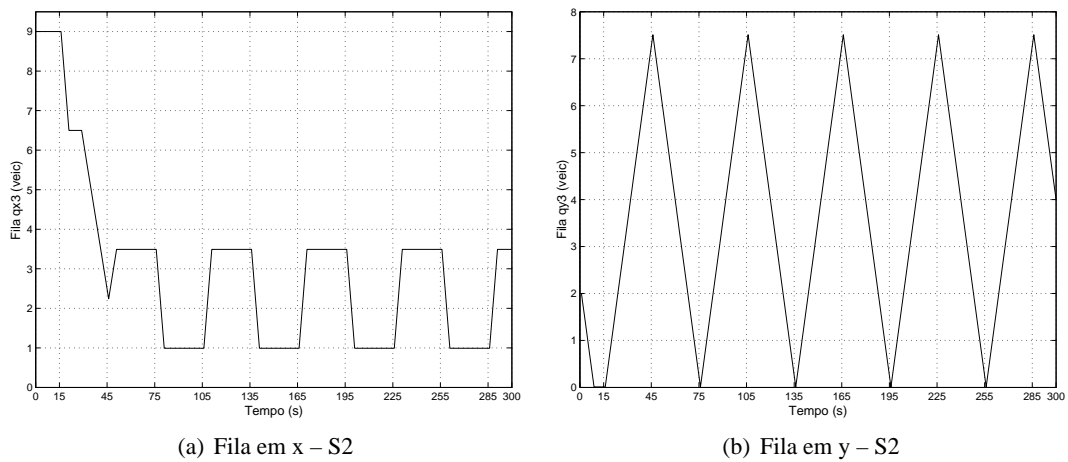


Figura 4.28: Comportamento das filas no semáforo S2 – Simulação 2

O gráfico mostra que, em regime permanente, os estados visitados nos autômatos S1 e S2 são os mesmos da simulação anterior. O fato de S1 oscilar entre os estados $u_1 = 1$ e $u_1 = 3$ da Figura 4.21, mesmo esgotando as filas q_{x1} e q_{y1} se deve ao fato disto acontecer nos instantes das trocas semafóricas. A mesma justificativa se aplica para o caso da fila q_{y3} .

O comportamento dos relógios r_1 e r_2 está na Figura 4.30. O relógio r_1 é reinicializado em 0 a cada novo ciclo de S1, e a variável r_2 está sempre no seu valor máximo em regime permanente. Apenas no primeiro ciclo r_2 tem valor inferior ao tempo de verde, e isto ocorre porque a fila em

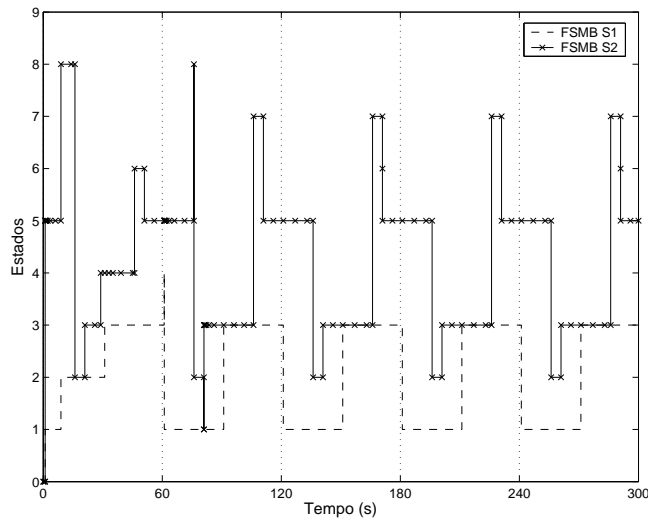


Figura 4.29: Evolução dos estados nos FSMBs – Simulação 2

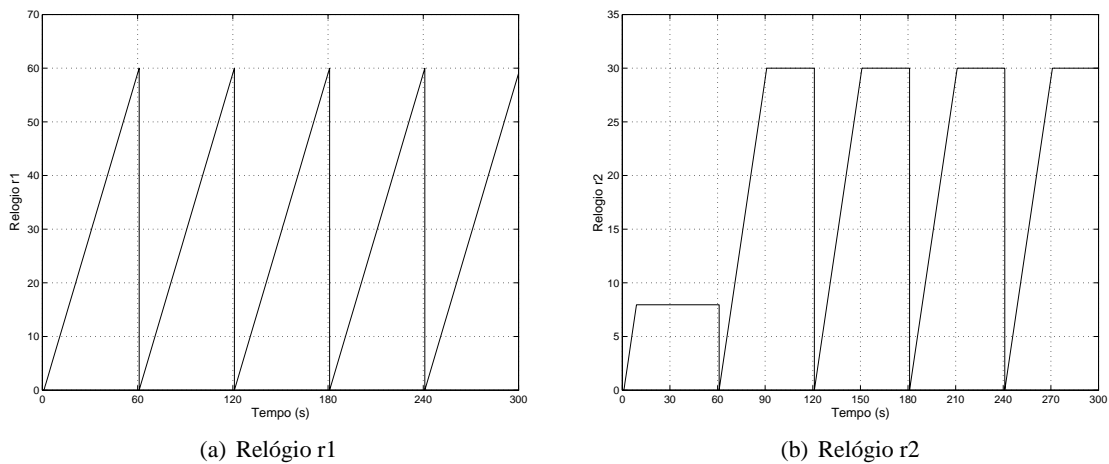


Figura 4.30: Comportamento dos relógios – Simulação 2

x_1 se esgota e é preciso armazenar o instante da troca de dinâmicas. A troca desta dinâmica fica comprovada com a visita ao estado $u_1 = 2$.

No primeiro ciclo de S2 a fila q_{x3} possui um comportamento diferente da simulação anterior. A fila permanece com um valor constante até o momento da abertura do sinal verde no *link* x_3 , quando se inicia uma descarga da fila (estado $u_2 = 2$). Quando os veículos que partiram de S1 alcançam o *link* x_3 o autômato transita para o estado $u_2 = 3$, onde a o fluxo de chegada e partida de veículos é f_{x1} . A ocorrência da troca de dinâmica no modo com-fila no semáforo S1 altera o comportamento da fila q_{x3} . Quando os veículos que partem de S1 com fluxo u_{x1} chegam em S2 o autômato S2 transita para o estado $u_2 = 4$, onde a fluxo de chegada é menor que o fluxo de partida. A fila é descarregada até o momento da abertura do sinal vermelho no *link* x_3 , quando a fila começa novamente a crescer. O fato da transição para o sinal vermelho em x ocorrer com fluxo de chegada igual u_{x1} justifica a visita ao estado $u_2 = 6$. Após o primeiro ciclo os fluxos de entrada e partida de veículos são sempre iguais, como mostra a situação de regime.

4.5 Verificação de Propriedades

Através das simulações apresentadas foi possível avaliar o comportamento das filas do sistema através da modelagem com autômatos híbridos. Além de possibilitar a análise da evolução do sistema ao longo do tempo, as simulações permitiram verificar que o acoplamento dos sistemas através das guardas funcionou corretamente. Entretanto, o objetivo deste trabalho também é realizar verificação formal dos sistemas de tráfego urbano modelados através dos sistemas híbridos.

Infelizmente, um problema que se encontra nas ferramentas de modelagem e verificação de sistemas híbridos é a complexidade computacional na busca do espaço de estados, e isto não é diferente com o *CheckMate*. Um sistema formado com duas interseções conectadas possui 6 variáveis de estado e a sua verificação não é viável. Entretanto, tenta-se verificar algumas propriedades relacionadas a um sistema composto por uma interseção isolada. O *CheckMate* permite modelar especificações escritas em lógica ACTL, e através delas é possível verificar situações de bloqueio, verificar se uma fila alcança ou não um dado tamanho no futuro e se determinada dinâmica contínua vai ser habilitada em algum momento, entre outros.

Nesta tese são apresentados resultados para duas especificações: especificação que verifica se o sistema é bloqueante; e especificação que verifica se existe tempo de verde ocioso.

4.5.1 Verificação 1 – Bloqueio

O modelo refinado de um sistema de tráfego como da Figura 4.6(a) (exemplo desenvolvido na seção 4.3.1) pode ser definido utilizando três variáveis de estado, duas para modelar as filas e uma para modelar o relógio que controla a divisão do tempo de ciclo entre os estágios. O autômato híbrido implementado no *CheckMate* para este sistema pode ser observado na Figura 4.31. As guardas implementadas estão na Tabela 4.7. Os estados *green – x1* e *green – x2* representam os estados onde o sistema está verde na direção *x* (estágio 2) e *red – x1* e *red – x2* os estados quando está vermelho na direção *x* (estágio 1). O estado *avoid* foi introduzido para fins de verificação de propriedades, sendo que o sistema transita para este estado quando as filas alcançam determinado valor.

Etiqueta	Guarda
$g1_1$	$q_{x1} = 0$
$g2_1$	$q_{y1} = 0$
$g3_1$	$r = cT$
$g4_1$	$r = T ; r := 0$
g_p	$q_{x1} > 50$
g_f	$q_{x1} > 50$

Tabela 4.7: Guardas do autômato para uma interseção isolada

A propriedade a ser verificada busca garantir que as filas nas direções *x* e *y* não ultrapassem o valor limite de 50 veículos, ou seja, verificar que o estado *avoid* nunca é visitado. Os parâmetros e

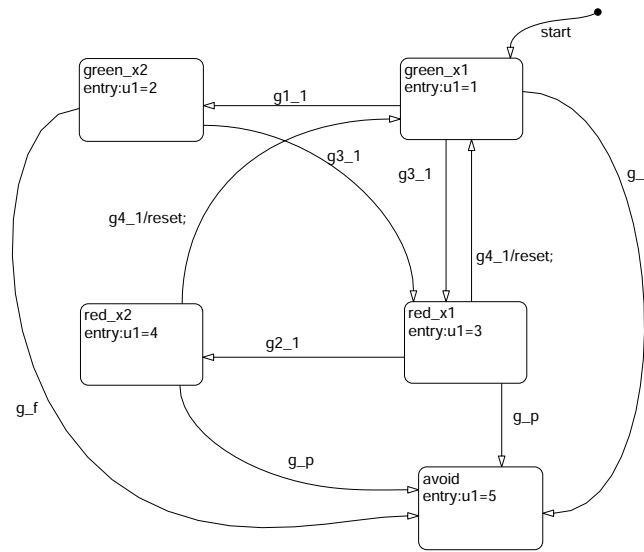


Figura 4.31: FSMB que modela semáforo para interseção isolada

condições iniciais do sistema para esta verificação são:

- tempo de ciclo $T = 90$ (segundos);
- *split* $c = 0.6$ (54 segundos estágio 2 e 36 segundos estágio 1);
- fluxo de saturação $f_{x1} = 3600/3600$ (veículos/segundo);
- fluxo de saturação $f_{y1} = 3600/3600$ (veículos/segundo);
- fluxo de entrada de veículos $u_{x1} = 2160/3600$ e $u_{y1} = 1440/3600$ (veículos/segundo);
- condições iniciais: $q_{x1} = 2$, $q_{y1} = 0$ e $r = 0$, além disso o semáforo S1 está com o sinal verde direção x;
- tempo de simulação $t = 600$ (segundos).

Os resultados de simulação apresentados na Figura 4.32 indicam que o estado avoid nunca é visitado pois as filas não atingem o valor limite. Entretanto, para garantir que isto nunca vai acontecer em nenhum momento do futuro é preciso realizar o processo de verificação.

O resultado obtido através do procedimento *verify* do *CheckMate* está a seguir, onde é possível garantir que o sistema nunca entrará no estado avoid com as condições iniciais utilizadas. A especificação $((AG \sim \text{out-of-bound}) \ \& \ (AG \sim \text{sem-1} == \text{avoid}))$ verifica que o sistema nunca sai dos limites da região de análise e que o estado avoid do autômato sem-1 nunca é visitado.

```
>> verify
Performing PIHA conversion.
Compiling threshold hyperplanes.
```

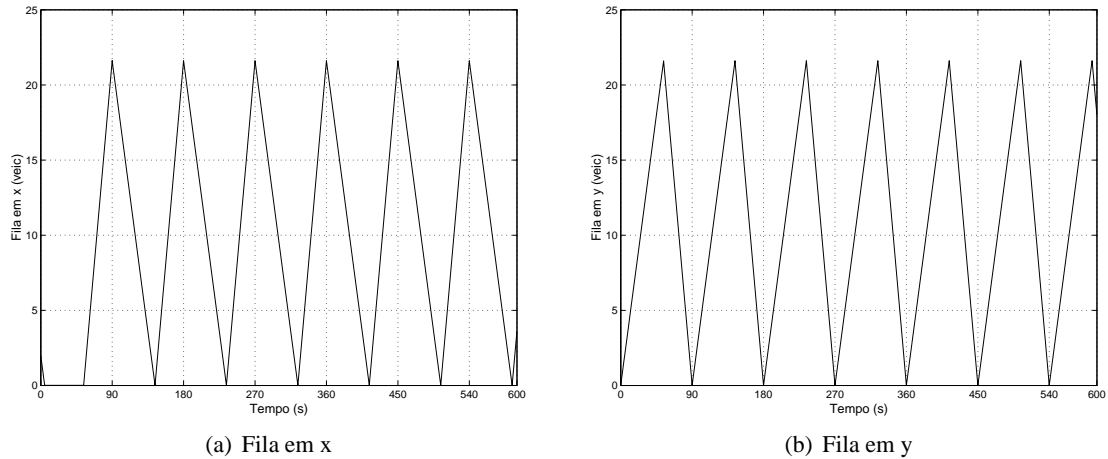


Figura 4.32: Comportamento das filas – Verificação 1

```

PIHA conversion successful.
PIHA conversion completed in 2.31 seconds.
Generating generic transition system.
.
.
.
Parsing specification 1: (AG ~out_of_bound)&(AG ~sem_1 == avoid)
Compiling list of atomic propositions.
* out_of_bound
* sem_1_in_avoid
Making refinement decision.
System never enters the state "avoid"
total verification time is 13.50 seconds.
>>

```

Quando se altera o fluxo de chegada de veículos em uma das direções, no caso $u_{y1} = 1500/3600$, é possível perceber que o comportamento da fila é alterado na direção y . A Figura 4.33 mostra que a fila na direção y não se esgota ao longo do tempo. Ao tentar verificar a propriedade verificada anteriormente se encontra o seguinte resultado:

```

>> verify
Performing PIHA conversion.
Compiling threshold hyperplanes.
PIHA conversion successful.
PIHA conversion completed in 2.27 seconds.
Generating generic transition system.
.
.
.

```

```

Parsing specification 1: (AG ~out_of_bound)&(AG ~sem_1 == avoid)
Compiling list of atomic propositions.
* out_of_bound
* sem_1_in_avoid
Making refinement decision.
System does not satisfy the specification of nevering entering the
state "avoid"
....
>>

```

O resultado da verificação diz que não é possível garantir que o sistema nunca entrará no estado avoid, o que já era esperado. A verificação é necessária porque não é possível afirmar que determinada situação vai sempre ou nunca acontecer através de resultados de simulação.

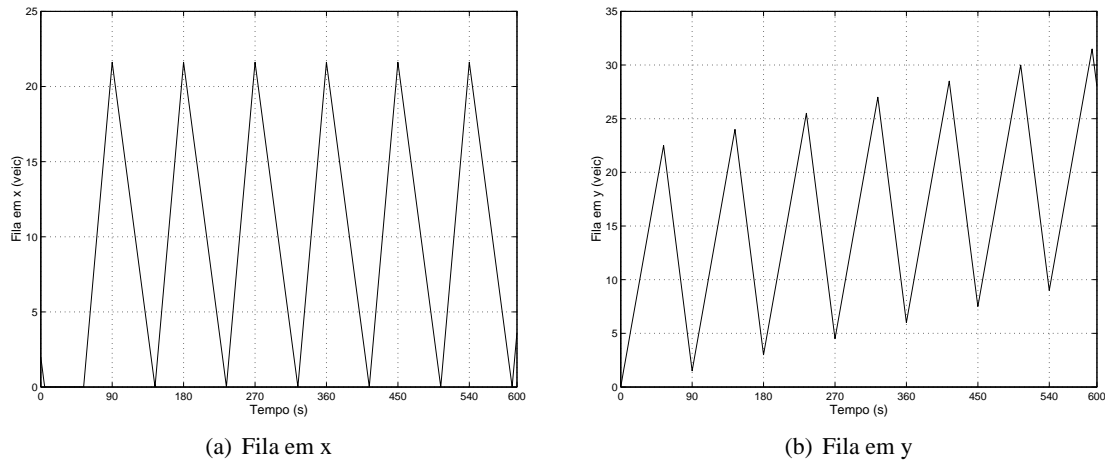


Figura 4.33: Comportamento das filas – Verificação 1 com fluxos alterados

4.5.2 Verificação 2 – Tempo Ocioso de Verde

Para realizar esta verificação foi adicionado um novo estado ao autômato híbrido que modela uma interseção isolada. O novo autômato é apresentado na Figura 4.34. A introdução do estado reach se deve ao desejo de modelar a situação onde, caso a fila q_{x1} permaneça nula por um período de tempo determinado, a indicação semafórica é alterada, aumentando assim o tempo de verde na direção y. Para poder realizar o controle do tempo que a fila q_{x1} permanece nula foi introduzida uma nova variável de estado, no caso o relógio r_2 . A guarda lim é satisfeita quando $r_2 > 10$, e g_f quando $q_{x1} > 150$ ou $q_{y1} > 150$.

Os parâmetros utilizados para esta simulação e verificação foram:

- tempo de ciclo $T = 90$ (segundos);
- $split\ c = 0.6$ (54 segundos estágio 2 e 36 segundos estágio 1);

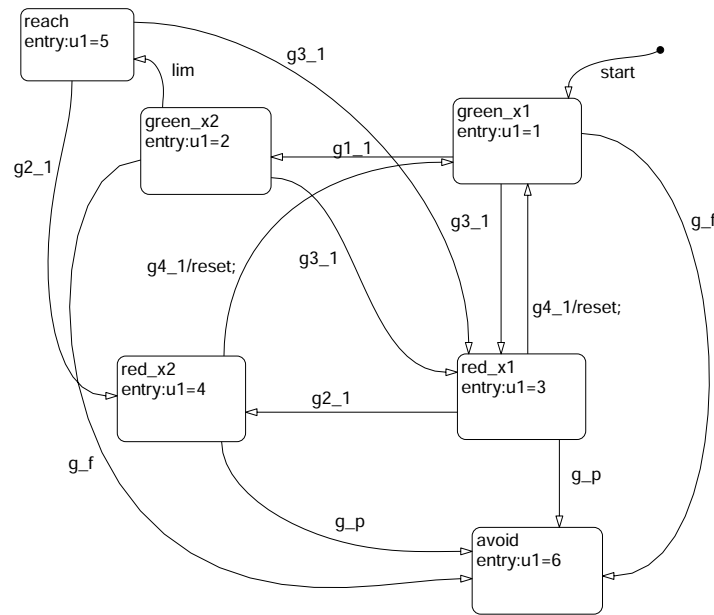


Figura 4.34: FSMB modificada que modela semáforo para interseção isolada

- fluxo de saturação $f_{x1} = 3600/3600$ (veículos/segundo);
- fluxo de saturação $f_{y1} = 3600/3600$ (veículos/segundo);
- fluxo de entrada de veículos $u_{x1} = 1650/3600$ e $u_{y1} = 2000/3600$ (veículos/segundo);
- condições iniciais: $q_{x1} = 10$, $q_{y1} = 0$, $r = 0$ e $r_2 = 0$, além disso o semáforo S1 está com o sinal verde direção x ;
- tempo de simulação $t = 500$ (segundos).

O comportamento da filas para esta simulação pode ser analisado na Figura 4.35. É possível perceber a fila q_{x1} é nula por mais de 10 segundos, fazendo com que o semáforo transite para o sinal verde na direção y em $t = 32$ segundos. Caso o estado `reach` não tivesse sido adicionado esta transição só aconteceria em $t = 54$ segundos e a fila q_{y1} não teria sido esgotada.

Neste caso deseja-se verificar duas propriedades: se o sistema entra no estado `avoid`, e neste caso vai bloquear pois atingiu um valor limite para alguma das filas; e se o sistema entra no estado `reach` em algum momento. Os resultados obtidos estão a seguir:

```
>> verify
Performing PIHA conversion.
Compiling threshold hyperplanes.
PIHA conversion successful.
.
.
.
```

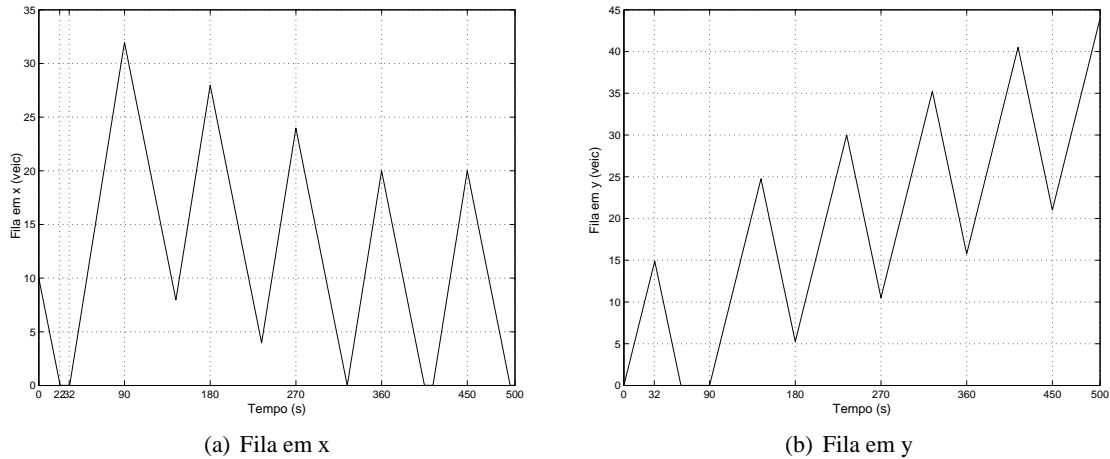



Figura 4.35: Comportamento das filas – Verificação 2

```
Parsing specification 1: (AG ~sem_1 == avoid)
```

```
Compiling list of atomic propositions.
```

```
* sem_1_in_avoid
```

```
Parsing specification 2: (AF sem_1 == reach)
```

```
Compiling list of atomic propositions.
```

```
* sem_1_in_reach
```

```
System does not satisfy specification 1.
```

```
Computing the "to-be-refined" (TBR) set.
```

```
System already satisfies specification 2. No refinement necessary.
```

```
>>
```

O resultado obtido confirma que a fila q_{y1} vai crescer até o momento que transitar para o estado avoid, bloqueando o sistema. Além disso, o resultado garante que em algum momento o estado reach também é visitado.

4.6 Conclusão

Neste capítulo foi apresentada uma metodologia de modelagem e análise para sistemas de tráfego urbano através da abordagem de sistemas híbridos, onde o formalismo utilizado para modelagem foi o autômato híbrido. A metodologia foi desenvolvida com o auxílio de exemplos, tentando facilitar a sua compreensão, e resultados de simulação e verificação também foram discutidos. Os resultados obtidos também podem ser encontrados em Garcia e Cury [2004a,b].

Os exemplos foram implementados com a ajuda de uma ferramenta computacional, o *CheckMate*, entretanto a ferramenta possui limitações quanto à verificação de sistemas híbridos definidos com

mais que 4 variáveis de estado. Isto ocorre devido à complexidade de calcular o conjunto de estados alcançáveis no processo de verificação. Esta ferramenta foi escolhida por já ter sido utilizada em outros trabalhos do grupo de pesquisa, entretanto, o problema de verificação não pode ser resolvido em sistemas com mais que 4 variáveis de estado.

O problema de síntese de controle não foi abordado neste capítulo devido ao grande número de variáveis de estado utilizadas no processo de modelagem do sistema. Em alguns casos o problema de controle de sistemas híbridos é traduzido em um problema equivalente de controle supervisorio de SEDs [Cury e Krogh, 1999], entretanto, é necessário obter um autômato discreto que modele o comportamento discreto do sistema. O número de variáveis de estado utilizadas nos modelos aqui desenvolvidos impossibilita a obtenção deste autômato discreto. Sendo assim, o problema de síntese de controle é definido apenas quando os sistemas de tráfego urbano são estudados como sistemas puramente discretos.

Como dito anteriormente, um formalismo de SEDs que se mostra eficiente para modelar sistemas que precisam ser sincronizados é a álgebra max-plus [Bacelli et al., 1992]. Este formalismo é apresentado no Capítulo 5, onde diversos conceitos formais necessários para compreender o Capítulo 6 são abordados.

Capítulo 5

Álgebra Max-Plus

No capítulo anterior os sistemas de tráfego urbano foram modelados como sistemas híbridos, onde o formalismo utilizado foi o autômato híbrido. A partir deste momento, a abordagem utilizada é puramente discreta, ou seja, os possíveis comportamentos contínuos do sistema não são modelados.

Atualmente existem vários formalismos que permitem modelar SEDs. O formalismo adotado neste trabalho é álgebra max-plus, devido à natureza do problema estudado, que é a sincronização de semáforos.

Este capítulo inicia-se com uma introdução informal da álgebra max-plus, através de um exemplo. Após, serão apresentados os formalismos matemáticos necessários para a proposta de modelagem e controle de sistemas de tráfego urbano. Para um aprofundamento ao estudo da álgebra max-plus sugere-se a leitura de [Baccelli et al., 1992].

5.1 Exemplo de um Sistema Dinâmico Max-Plus Linear

O problema de sincronização de eventos é facilmente detectado em um sistema de trens, quando se deseja coordenar os horários de chegada e partida dos trens para prover um melhor aproveitamento do sistema pelos usuários. O exemplo apresentado em Heidergott et al. [2006], baseado em um pequeno sistema de trens, é bastante útil para apresentar um sistema dinâmico max-plus linear, onde a álgebra max-plus pode ser introduzida de maneira natural.

Considere uma rede de trens simples entre duas cidades, cada uma com uma estação. As estações são denominadas E_1 e E_2 , e são conectadas através de duas vias férreas, como ilustra a Figura 5.1.

Uma via tem o sentido $E_1 \rightarrow E_2$ e o tempo de viagem de um trem é de 3 unidades de tempo. A outra via tem o sentido inverso e o tempo de viagem é de 5 unidades de tempo. Juntas, estas linhas formam um circuito. Trens vindo de E_1 e chegando em E_2 devem retornar para E_1 através da outra via, e trens que iniciaram em E_2 irão, após visitar E_1 , voltar para E_2 . Além disso, existem outros dois circuitos, que representam as vias que conectam os trens das cidades com as estações E_1 e E_2 . Uma

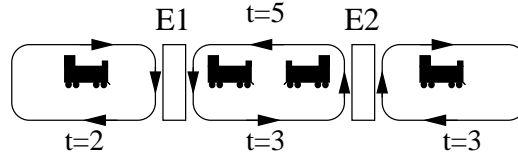


Figura 5.1: Rede de trens entre duas cidades

viagem através do circuito da primeira cidade (esquerdo da Figura 5.1) leva 2 unidades de tempo e no circuito da segunda cidade 3 unidades de tempo. As estações locais das cidades não são modeladas. Deseja-se uma escala de horários dos trens sujeitos aos seguintes critérios, restrições e hipóteses de funcionamento:

- os tempos de viagem são fixos;
- a frequência dos trens deve ser a mais alta possível;
- os trens chegando em uma estação devem esperar pelo outro trem para permitir a troca de passageiros;
- os trens devem partir da estação o mais cedo possível.

Assume-se que existem 4 trens no modelo, um para cada circuito mais externo e dois trens para o circuito mais interno. Os eventos do modelo são as partidas dos trens, onde a variável x_1 indica o tempo de partida dos dois trens que partem de E_1 , um na direção de E_2 e outro em direção ao circuito que volta a E_1 , e eles partem ao mesmo tempo devido à troca de passageiros. A variável x_2 representa, de forma similar, o tempo de partida dos trens na estação E_2 . Os instantes de partida são escritos como um vetor $x \in \mathbb{R}^2$, onde a primeira partida do dia é dada por $x(0)$. O k -ésimo instante de partida é indicado por $x(k)$.

Sabe-se que os instantes de partida tem que respeitar os critérios definidos para o sistema. Pode-se expressar os instantes de partida x_1 através de duas inequações:

$$x_1(k+1) \geq x_1(k) + 2 \quad (5.1)$$

$$x_1(k+1) \geq x_2(k) + 5. \quad (5.2)$$

As inequações (5.1) e (5.2) indicam que a $k+1$ -ésima partida de x_1 vai depender do instante de partida anterior dos trens envolvidos e dos tempos de viagem entre as estações. Similarmente definem-se as inequações para os instantes de partida x_2 :

$$x_2(k+1) \geq x_1(k) + 3 \quad (5.3)$$

$$x_2(k+1) \geq x_2(k) + 3. \quad (5.4)$$

Para um trem partir as duas condições devem ser respeitadas, ou seja, ele vai partir no instante de tempo mais tardio entre as duas inequações (operador max). Além disso, deseja-se que a frequência de partida seja a mais alta possível e que os trens partam o mais cedo possível, então é possível escrever as seguintes equações:

$$x_1(k+1) = \max(x_1(k) + 2, x_2(k) + 5)) \quad (5.5)$$

$$x_2(k+1) = \max(x_1(k) + 3, x_2(k) + 3)). \quad (5.6)$$

Neste caso se o instante de partida inicial $x(0)$ é conhecido, todos os instantes futuros são possíveis de serem calculados. Uma generalização direta das equações para o caso de n estações é dada por:

$$x_i(k+1) = \max(x_1(k) + a_{i1}, x_2(k) + a_{i2}, \dots, x_n(k) + a_{in}) \quad (5.7)$$

para $i = 1, 2, \dots, n$ onde a_{in} denota o tempo de viagem entre as estações i e n . Quando não existe ligação entre E_i e E_n define-se que o tempo de viagem é $a_{in} = -\infty$.

Se se utilizar a notação da álgebra max-plus a equação (5.7) é expressa da seguinte forma:

$$x_i(k+1) = (x_1(k) \otimes a_{i1}) \oplus (x_2(k) \otimes a_{i2}) \oplus \dots \oplus (x_n(k) \otimes a_{in}) \quad (5.8)$$

onde os operadores \oplus e \otimes são denominados *max* e *plus*, respectivamente, sendo $\oplus = \max$ e $\otimes = +$. É possível expressar a equação (5.8) matricialmente como

$$x(k+1) = A \otimes x(k), \quad (5.9)$$

onde as linhas da matriz correspondem as variáveis x_i e os tempos de viagem da estação j à estação i são os elementos a_{ij} da matriz.

As equações (5.7), (5.8) e (5.9) são equivalentes. Entretanto, além de representar o sistema através de equações é possível representá-lo através de um formalismo com representação gráfica, no caso um grafo dirigido com n nós, onde cada nó corresponde a uma variável x_i e os arcos correspondem às ligações onde $a_{ik} \neq -\infty$. Para o caso do exemplo dos trens é possível obter o grafo da Figura 5.2.

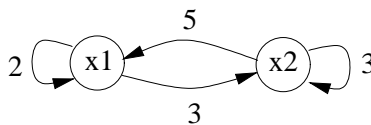


Figura 5.2: Grafo para rede de trens entre duas cidades

Além do grafo é possível representar a equação (5.8) através de uma rede de Petri temporizada. Uma definição para as redes de Petri temporizadas, extraída de Heidergott et al. [2006] encontra-se a seguir.

Definição 5.1 (Rede de Petri Temporizada). *Uma rede de Petri temporizada \mathcal{R} é caracterizada por $\mathcal{P}, \mathcal{Q}, \mathcal{D}, \mathcal{M}_0$ e \mathcal{T} , onde:*

- \mathcal{P} é um conjunto finito de lugares denotados por $p_i, i = 1, 2, \dots, |\mathcal{P}|$;
- \mathcal{Q} é um conjunto finito de transições denotadas por $q_j, j = 1, 2, \dots, |\mathcal{Q}|$;
- \mathcal{D} é o conjunto de arcos $(p_i \rightarrow q_j)$, que ligam os lugares às transições, e $(q_j \rightarrow p_i)$, que ligam as transições aos lugares;
- \mathcal{M}_0 é a marcação inicial;
- \mathcal{T} é um vetor com os tempos de espera associados aos lugares p_i .

As redes de Petri que são utilizadas neste trabalho são grafos de eventos temporizados (GETs), que são uma subclasse das redes de Petri temporizadas [Maia, 2003].

Definição 5.2 (Grafo de eventos Temporizado – GET). *Um grafo de eventos temporizado é uma rede de Petri temporizada na qual cada lugar p_i tem um tempo de espera associado e somente uma transição de entrada e somente uma transição de saída.*

Uma transição de saída única significa que todos os conflitos potenciais no uso de recursos foram previamente resolvidos por alguma política. Já uma transição de entrada única significa que não há concorrência ao consumo ou fornecimento de recursos no grafo de eventos temporizados. Os GET podem representar o comportamento dos sistemas modelados através da álgebra max-plus, ou seja, representam as equações que descrevem estes sistemas.

A Figura 5.3 apresenta um GET para o exemplo dos trens. O conjunto $P = \{p_1, p_2, p_3, p_4\}$ modela os lugares da rede, indicando os estados que o sistema pode estar, e o conjunto $T = \{x_1, x_2\}$ as transições possíveis entre os lugares. Associados aos lugares estão as fichas (círculos pretos), que definem a marcação da rede, e no caso da Figura 5.3 também estão associados valores que indicam o tempo mínimo que o sistema deve permanecer naquele estado. Uma transição só pode acontecer quando os seus lugares de entrada estão marcados com uma ficha, e esta ficha se desloca para os lugares de saída da transição.

Através da rede de Petri é possível perceber a sincronização entre os trens para calcular a tabela de horários. Observamos que as transições que representam as variáveis x_1 e x_2 só podem ser disparadas após respeitar os tempos de viagem envolvidos, estando os tempos de viagem e os trens (fichas) relacionados aos lugares da rede.

No exemplo descrito não se considera a existência de ações de controle, ou seja, o sistema é modelado como um sistema autônomo. No entanto, no caso mais geral, sistemas como o descrito aqui podem ser representados por equações lineares do tipo:

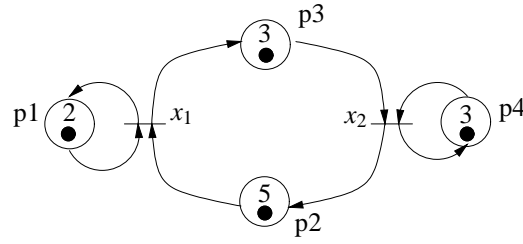


Figura 5.3: GET para a rede de trens entre duas cidades

$$x(k) = [A \otimes x(k-1)] \oplus [B \otimes u(k)]. \quad (5.10)$$

O vetor $x(k)$ representa o estado do sistema e $u(k)$ uma entrada externa, e correspondem ao k -ésimo disparo das transições representadas por estas variáveis. Assim como a multiplicação na álgebra convencional, o operador \otimes costuma ser omitido nas equações, logo a equação costuma ser expressa como:

$$x(k) = Ax(k-1) \oplus Bu(k). \quad (5.11)$$

A definição formal da álgebra max-plus, bem como os aspectos necessários para compreensão deste trabalho serão apresentados na próxima seção.

5.2 Teoria de Dióides

As definições apresentadas a seguir foram extraídas de Baccelli et al. [1992], Maia [2003], Maia et al. [2005a], Heidergott et al. [2006], Amari [2005].

Definição 5.3 (Semi-anel). *Um semi-anel é um conjunto \mathcal{D} equipado com duas leis de composição interna \oplus e \otimes , chamadas soma e multiplicação, que verificam os seguintes axiomas:*

1. *Associatividade da adição:*

$$\forall a, b, c \in \mathcal{D}, (a \oplus b) \oplus c = a \oplus (b \oplus c).$$

2. *Comutatividade da adição:*

$$\forall a, b \in \mathcal{D}, a \oplus b = b \oplus a.$$

3. *Associatividade da multiplicação:*

$$\forall a, b, c \in \mathcal{D}, (a \otimes b) \otimes c = a \otimes (b \otimes c).$$

4. *Distributividade da multiplicação em relação à adição:*

$$\begin{aligned}\forall a, b, c \in \mathcal{D}, a \otimes (b \oplus c) &= (a \otimes b) \oplus (a \otimes c), \\ (b \oplus c) \otimes a &= (b \otimes a) \oplus (c \otimes a).\end{aligned}$$

5. *Existência do elemento neutro da adição:*

$$\forall a \in \mathcal{D}, a \oplus \varepsilon = a.$$

6. *Absorção pelo elemento neutro da adição:*

$$\forall a \in \mathcal{D}, \varepsilon \otimes a = a \otimes \varepsilon = \varepsilon.$$

7. *Existência do elemento identidade da multiplicação:*

$$\forall a \in \mathcal{D}, a \otimes e = e \otimes a = a.$$

O que diferencia um semi-anel de um anel é a ausência do elemento inverso aditivo. Além disso, quando um semi-anel possui a propriedade de idempotência da adição, isto é, $\forall a \in \mathcal{D}, a \oplus a = a$, ele é chamado de dióide [Baccelli et al., 1992] ou semi-anel idempotente.

Definição 5.4. Um dióide \mathcal{D} é dito comutativo se a multiplicação é comutativa.

Exemplos de dióides comutativos são a álgebra max-plus e min-plus.

Definição 5.5 (Álgebra max-plus). A álgebra max-plus é um caso particular de um dióide. Definem-se $\varepsilon = -\infty$ e $e = 0$, e denota-se por \mathbb{R}_{\max} o conjunto $(\mathbb{R} \cup \{\varepsilon\}, \oplus, \otimes)$, onde $\oplus = \max$ e $\otimes = +$.

Definição 5.6 (Álgebra min-plus). Na álgebra min-plus definem-se $\varepsilon = +\infty$ e $e = 0$, e denota-se por \mathbb{R}_{\min} o conjunto $(\mathbb{R} \cup \{\varepsilon\}, \oplus, \otimes)$, onde $\oplus = \min$ e $\otimes = +$.

Tanto a álgebra max-plus quanto a min-plus podem ser definidas sobre o conjunto dos inteiros, sendo $\mathbb{Z}_{\max} = (\mathbb{Z} \cup \{\varepsilon\}, \max, +)$ e $\mathbb{Z}_{\min} = (\mathbb{Z} \cup \{\varepsilon\}, \min, +)$.

Um exemplo de dióide não-comutativo é o dióide matricial.

Definição 5.7 (Dióide matricial). O conjunto das matrizes de $\mathbb{Z}_{\max}^{n \times n}$ é um dióide matricial no qual a soma e o produto de duas matrizes $A, B \in \mathbb{Z}_{\max}^{n \times n}$ são definidos por:

$$\begin{aligned}[A \oplus B]_{ij} &= a_{ij} \oplus b_{ij} \\ [A \otimes B]_{ij} &= \bigoplus_{k=1}^n a_{ik} \otimes b_{kj}.\end{aligned}$$

Sejam A e B matrizes $\in \mathbb{Z}_{\max}^{n \times n}$.

$$A = \begin{pmatrix} 2 & 3 \\ 5 & 6 \end{pmatrix} \quad B = \begin{pmatrix} 8 & 4 \\ 6 & 2 \end{pmatrix}$$

A soma e produto das matrizes A e B são dados pelas seguintes matrizes $\in \mathbb{Z}_{\max}^{n \times n}$:

$$A \oplus B = \begin{pmatrix} 8 & 4 \\ 6 & 6 \end{pmatrix} \quad A \otimes B = \begin{pmatrix} 10 & 6 \\ 13 & 9 \end{pmatrix}$$

A potência de uma matriz $A \in \mathbb{Z}_{\max}^{n \times n}$ é definida como:

$$A^{\otimes k} = \underbrace{A \otimes \dots \otimes A}_k = \bigotimes_{i=1}^k A$$

onde $A^0 = I$ é a matriz identidade.

Similarmente define-se a potência de um escalar c como:

$$c^{\otimes k} = k \times c, \quad k \in \mathbb{Z}.$$

Definição 5.8 (Dióide completo). *Um dióide é dito “completo” se ele é fechado em relação a somas infinitas e se a operação \otimes se distribui sobre as somas infinitas, ou seja, para todo $c \in \mathcal{D}$ e todo sub-conjunto $A \subseteq \mathcal{D}$,*

$$c \otimes \left(\bigoplus_{x \in A} x \right) = \bigoplus_{x \in A} c \otimes x.$$

O dióide \mathbb{Z}_{\max} completado pelo elemento $+\infty$ é um dióide completo, denotado por $\overline{\mathbb{Z}}_{\max} = (\mathbb{Z} \cup \{-\infty, +\infty\}, \max, +)$. Uma importante operação definida em qualquer dióide é a operação estrela de Kleene, definida por

$$a^* = \bigoplus_{i \in \mathbb{N}} a^i$$

com $a^i = a \otimes a^{(i-1)}$ e $a^0 = e$.

Definição 5.9 (Relação de ordem de um dióide). *A propriedade de idempotência da lei aditiva \oplus induz uma relação de ordem para um dióide \mathcal{D} . A relação é dada por \preceq , e é definida por:*

$$\forall a, b \in \mathcal{D}, a \preceq b \Leftrightarrow a \oplus b = b.$$

Esta relação de ordem é compatível com as leis \oplus e \otimes , sendo:

$$\begin{aligned} \forall a, b, c \in \mathcal{D}, a \preceq b &\Rightarrow a \oplus c \preceq b \oplus c, \\ a \preceq b &\Rightarrow a \otimes c \preceq b \otimes c. \end{aligned}$$

No dióide $(\mathbb{R} \cup \{+\infty, -\infty\}, \max, +)$ a relação \preceq é uma relação de ordem que corresponde a ordem usual \leq ,

$$a \preceq b \Leftrightarrow b = \max(a, b) \Leftrightarrow a \leq b.$$

Entretanto, no dióide $(\mathbb{R} \cup \{+\infty, -\infty\}, \min, +)$ a relação \preceq corresponde a ordem \geq ,

$$a \preceq b \Leftrightarrow b = \min(a, b) \Leftrightarrow a \geq b.$$

5.2.1 Modelagem de GET através de Dióides

O comportamento dinâmico dos GET é usualmente descrito por equações não-lineares que envolvem os operados \max e $+$ (ou \min e $+$) na álgebra convencional. Entretanto, este comportamento não-linear pode ser linearizado quando se utiliza a álgebra de dióides.

Para um GET, cada transição representa uma determinada família de eventos de mesma natureza e a ocorrência de um evento corresponde ao disparo de uma transição. Os GET podem ser descritos por datadores ou contadores. Os GET da Figura 5.4, extraída de Maia [2003], podem ser descritos pelos seguintes datadores e contadores.

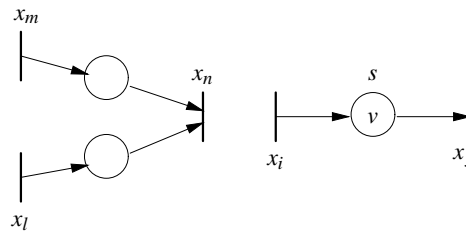


Figura 5.4: Configurações para transições de GET

Descrição por datadores: Seja $x_i(k)$, $x_m(k)$ e $x_l(k)$ a data do k -ésimo disparo das transições x_i , x_m e x_l , respectivamente, e sejam v o número de fichas e s o tempo de espera de um lugar. A partir da Figura 5.4 é possível escrever as seguintes equações:

$$\begin{aligned} x_n(k) &\geq \max(x_m(k), x_l(k)) \\ x_j(k) &\geq x_i(k - v) + s. \end{aligned}$$

Descrição por contadores: Seja $x_i(t)$, $x_m(t)$ e $x_l(t)$ o número de disparos de x_i , x_m e x_l , respectivamente, ocorridos até a data t , tem-se que:

$$\begin{aligned} x_n(t) &\leq \min(x_m(t), x_l(t)) \\ x_j(t) &\leq x_i(t - s) + v. \end{aligned}$$

A descrição por datador significa que o k -ésimo disparo da transição x_n só pode acontecer após o k -ésimo disparo das transições x_m e x_l . Já a descrição por contadores significa que na data t o número

de disparos da transição x_n é sempre menor ou igual ao número de transições x_m e x_l . As equações também mostram que o número de fichas em um dado lugar ocasiona um avanço nos disparos e que o tempo de espera ocasiona um retardo na data em relação às transições x_i e x_j .

Os operadores \max , \min e $+$ permitem a descrição linear do GET via álgebra de dióides. Desta forma, o comportamento dinâmico de datadores e contadores pode ser descrito de maneira natural utilizando os dióides \mathbb{Z}_{\max} e \mathbb{Z}_{\min} respectivamente.

5.2.2 Teoria da Residuação

A solução de equações do tipo $y = f(x)$ definidas sobre dióides pode apresentar um número infinito de soluções ou nenhuma solução, diferentemente do caso da álgebra tradicional. As operações \oplus e \otimes não são inversíveis, particularmente para aplicações matriciais, logo não é possível inverter aplicações definidas sob forma analítica sobre dióides. A Teoria da Residuação [Blyth e Janowitz, 1972] se ocupa da solução deste problema em conjuntos parcialmente ordenados. Através da teoria da residuação é possível definir pseudo-inversas, conseqüentemente definir soluções extremas para a inequação $y \succeq f(x)$. Alguns resultados baseados em Maia [2003]; Maia et al. [2005b] serão apresentados a seguir.

Definição 5.10 (Isotonia). *Uma aplicação f definida sobre um dióide $(\mathcal{D}, \oplus, \otimes)$ em um dióide $(\mathcal{C}, \oplus, \otimes)$ é chamado isotônica se para todo $a, b \in \mathcal{D}$ a seguinte relação de ordem é preservada*

$$a \preceq b \Rightarrow f(a) \preceq f(b).$$

Considere o mapeamento isôtonico $f : \mathcal{D} \rightarrow \mathcal{C}$, sendo \mathcal{D} e \mathcal{C} dióides completos. Se existir um elemento máximo x_{op} do conjunto $\{x \mid y \succeq f(x), x \in \mathcal{D}, y \in \mathcal{C}\}$ diz-se que x_{op} é o resíduo do mapeamento f em y . Se f tiver um resíduo em qualquer ponto $y \in \mathcal{C}$ o mapeamento f é dito residuável, sendo o resíduo denotado por $f^\sharp(y)$ [Baccelli et al., 1992]. Dualmente, o mínimo conjunto do $\{x \mid y \preceq f(x), x \in \mathcal{D}, y \in \mathcal{C}\}$, se ele existir, é denominado resíduo dual sendo denotado por $f^\flat(y)$ e o mapeamento f é dito dualmente residuável.

As funções $L_a(x) = a \otimes x$ e $R_a(x) = x \otimes a$ são residuáveis, sendo seus resíduos denotados respectivamente por $L_a^\sharp = a \backslash x$ (divisão à esquerda por a) e $R_a^\sharp = x \not\backslash a$ (divisão à direita por a). No caso particular do dióide $\overline{\mathbb{Z}}_{\max}$ tem-se que $L_a^\sharp = R_a^\sharp = x - a$ (operação de subtração na álgebra tradicional).

De maneira geral, se \mathcal{D} é um dióide completo, então $\mathcal{D}^{n \times n}$ também é um dióide e as operações $L_A(X) = A \otimes X$ e $R_A(X) = X \otimes A$ também são residuáveis, sendo A e X matrizes $n \times n$ com coeficientes em \mathcal{D} . A matriz $L_A^\sharp(X) = A \backslash X$ pode ser calculada por:

$$(A \backslash X)_{ij} = \bigwedge_{l=1}^n A_{li} \backslash X_{lj}$$

onde \bigwedge é o operador \min ($a \wedge b$ é o maior elemento do que a e menor do que b). A operação de residuação entre elementos das matrizes é a definida para os elementos do dióide \mathcal{D} . De modo

análogo, define-se o resíduo $X \setminus A$ substituindo na expressão anterior o operador \setminus pelo operador \setminus e invertendo as posições dos coeficientes das matrizes A e X .

5.2.3 Teoria Spectral de Matrizes no \mathbb{R}_{\max}

Para introduzir os resultados sobre a teoria spectral de matrizes no \mathbb{R}_{\max} é necessário relembrar algumas definições.

Um grafo dirigido G é definido por um conjunto de nós interconectados por arcos orientados. Se os arcos que ligam os nós i e j possuem pesos associados os grafos são denominados valorados. Este arco é denotado por (i, j) e o peso $w(i, j)$ deste arco é igual ao termo A_{ij} de uma matriz A associada ao grafo.

Um caminho em um grafo é uma seqüência de arcos $(p_0, p_1), (p_1, p_2), \dots, (p_{l-1}, p_l)$, também representado por $p = (p_0 \rightarrow p_1 \rightarrow \dots \rightarrow p_l)$, e o comprimento w do caminho p é dado pelo número de nós percorridos, denotado por $|p|_l$. O peso de um caminho p , denotado por $|p|_w$ é dado pela soma dos pesos de seus arcos. O peso médio é dado pela divisão do peso pelo comprimento:

$$\frac{|p|_w}{|p|_l}$$

Quando um caminho é fechado, ou seja $p_0 = p_l$, ele é denominado circuito. Um caminho elementar é aquele em que nenhum nó aparece mais que uma vez, ou seja, não contém um circuito. Se o caminho é um circuito o peso médio é chamado ciclo médio. Circuitos cujos ciclos médios sejam iguais ao máximo ciclo médio do grafo são chamados circuitos críticos.

Definição 5.11 (Grafo de precedência). *Seja $A \in \mathcal{D}^{n \times n}$ uma matriz quadrada com coeficientes no dióide \mathcal{D} . Denota-se $G(A)$ o grafo de precedência de A onde existem n nós e os arcos (i, j) são valorados com A_{ij} . Se $A_{ij} \neq \varepsilon$ existe um arco que liga os nós i e j , caso contrário a ligação não existe.*

Definição 5.12 (Grafo fortemente conexo). *Um grafo $G(A)$ é dito fortemente conexo se para todo par de nós $(i, j) \in G(A)$ o nó j é alcançável a partir do nó i .*

Definição 5.13 (Matriz irredutível). *Uma matriz $A \in \mathcal{D}^{n \times n}$ é considerada irredutível se o grafo de precedência $G(A)$ é fortemente conexo.*

As matrizes $n \times n$ com coeficientes definidos sobre os dióides apresentam algumas propriedades. É possível associar um grafo orientado e valorado, chamado grafo de precedência, a cada matriz quadrada, ou seja, existe uma representação gráfica para as matrizes. Além disso, se este grafo é fortemente conexo é possível afirmar que sua matriz correspondente é irredutível. Para uma matriz A irredutível existe um inteiro c que garante a seguinte propriedade de ciclicidade:

$$A^{k+c} = \lambda^c \otimes A^k$$

onde k é superior a um inteiro K e λ representa o autovalor da matriz A . Esta ciclicidade define um comportamento periódico em regime permanente, após um regime transitório [Amari, 2005].

Definição 5.14 (Autovalor e autovetor). *Seja $A \in \mathbb{R}_{\max}$ uma matriz quadrada, Se $\lambda \in \mathbb{R}_{\max}$ é um escalar e $v \in \mathbb{R}_{\max}$ um vetor que contém ao menos um elemento finito de modo que*

$$A \otimes v = \lambda \otimes v$$

então λ é chamada autovalor de A e v um autovetor de A associado ao autovetor λ .

Para o caso de matrizes irredutíveis a solução λ é única e calculada através da seguinte expressão:

$$\lambda = \bigoplus_{k=1}^n (\text{trace} A^k)^{1/k}$$

onde $\text{trace} A^k = \bigoplus_{i=1}^n (A^k)_{ii}$ e n corresponde a ordem da matriz A .

Além disso, em Baccelli et al. [1992] prova-se que se A é irredutível o autovalor também é dado pelo máximo ciclo médio de $G(A)$:

$$\lambda = \bigoplus_{i=1}^r \frac{|p|_w}{|p|_l}$$

onde r corresponde ao número de circuitos de $G(A)$.

5.3 Semimódulos (A,B)-Invariantes

Muitos resultados da teoria de controle clássico tem sua versão análoga em álgebra max-plus. Katz [2003, 2007] apresenta em seu trabalho a extensão da abordagem geométrica para teoria de sistemas dinâmicos lineares proposta por Wonham [1985]. O conceito de subespaço (A,B)-invariante apresentado por Wonham [1985] tem proporcionado muitas soluções para diversos problemas de controle.

Um sistema linear (na teoria de controle clássico) associado às matrizes (A, B, C) definidas sobre o conjunto dos reais \mathbb{R} é definido pelas seguintes equações:

$$\dot{x} = Ax + Bu \tag{5.12}$$

$$y = Cx \tag{5.13}$$

onde x define o estado do sistema, u uma entrada externa e y uma saída do sistema. A noção de subespaço (A,B)-invariante está relacionada com a equação (5.12).

Definição 5.15 (Subespaço (A,B)-invariante). *Um subespaço $\mathcal{V} \subset X$ é (A,B)-invariante se para qualquer que seja o estado inicial $x(0) \in \mathcal{V}$ existe um controle $u(t)$ tal que a trajetória resultante $x(t)$*

permaneça sempre em \mathcal{V} . Se $u(t)$ existe então ele pode ser implementado por uma realimentação de estados $F : X \rightarrow U$ que satisfaz $(A + BF)\mathcal{V} \subset \mathcal{V}$.

Para compreender o conceito de semimódulo (A,B) -invariante, utilizado nesta tese, são necessárias algumas definições, que foram extraídas de Katz [2007] e Katz [2003]. O primeiro passo é introduzir o conceito de semimódulo, que é o análogo em semi-anéis do espaço vetorial nos campos na teoria clássica.

Definição 5.16 (Semimódulo). *Um semimódulo sobre um semi-anel $(S, \oplus, \otimes, \epsilon_S, e)$ é um monóide comutativo $(X, \hat{\oplus})$, com elemento neutro ϵ_X e equipado com um mapa $S \times X \rightarrow X$, $(\lambda, x) \rightarrow \lambda \cdot x$ (ação à esquerda) que satisfaz:*

1. $(\lambda \otimes \mu) \cdot x = \lambda \cdot (\mu \cdot x)$,
2. $\lambda \cdot (x \hat{\oplus} y) = (\lambda \cdot x) \hat{\oplus} (\lambda \cdot y)$,
3. $(\lambda \oplus \mu) \cdot x = (\lambda \cdot x) \hat{\oplus} (\mu \cdot x)$,
4. $\epsilon_S \cdot x = \epsilon_X$,
5. $\lambda \cdot \epsilon_X = \epsilon_X$,
6. $e \cdot x = x$,

para todo $x, y \in X$ e $\lambda, \mu \in S$.

Definição 5.17 (Subsemimódulo). *Um subsemimódulo de X é um subconjunto $Z \subset X$ de modo que $\lambda x \hat{\oplus} \mu y \in Z$ para todo $x, y \in Z$ e $\lambda, \mu \in S$.*

Se $G \subset X$, denota-se por $\text{span}G$ o subsemimódulo de X gerado por G , que é o conjunto de todos os $x \in X$ para os quais existe um número finito de elementos u_1, \dots, u_k de G e um número finito de escalares $\lambda_1, \dots, \lambda_k \in S$ tal que $x = \hat{\oplus}_{i=1, \dots, k} \lambda_i u_i$. Finalmente, se $C \in S^{n \times r}$ denota-se por $\text{Im } C$ (imagem de C) o subsemimódulo de S gerado pelas colunas de C .

Define-se um semi-anel por (S, \oplus, \otimes) e a evolução de um sistema com coeficientes em S é determinada por um conjunto de equações na forma

$$x(k) = Ax(k-1) \oplus Bu(k) \quad (5.14)$$

onde $A \in S^{n \times n}$, $B \in S^{n \times q}$ e $x(k) \in S^{n \times 1}$, $u(k) \in S^{q \times 1}$ são, respectivamente, as seqüências de estado e os vetores de controle.

Definição 5.18 (Semimódulo geometricamente (A,B) -invariante). *Diz-se que um semimódulo $X \subset \mathbb{Z}_{\max}$ é (geometricamente) (A,B) -invariante se para todo $x \in X$ existe uma sucessão de vetores de controle que façam com que a trajetória do sistema (equação (5.14)) permaneça para sempre em X , quando estado inicial do sistema é $x \in X$.*

5.3.1 Equação $Ax = Bx$ sobre o \mathbb{R}_{\max}

O teorema seguinte foi demonstrado por [Butkovic e Hegedüs, 1984] e redescoberto por [Gaubert, 1992].

Teorema 5.1 (Teorema da finitude). *Sejam $A, B \in \mathbb{R}_{\max}^{n \times p}$. O conjunto $X \subset \mathbb{R}_{\max}^p$ das soluções do sistema max-plus linear $Ax = Bx$ é um semimódulo finitamente gerado, ou seja, um semimódulo gerado em um número finito de passos.*

Dado um semi-anel (S, \oplus, \otimes) arbitrário, chama-se hiperplano de S^q , para $q \in \mathbb{N}$ todo conjunto da forma $\{x \in S^q \mid ax = bx\}$, onde $a, b \in S^q$. Diz-se que um semi-anel S tem a propriedade da finitude se para todo $q \in \mathbb{N}$ os hiperplanos de S^q são semimódulos finitamente gerados. O Teorema 5.1 diz que o semi-anel max-plus tem a propriedade da finitude [Katz, 2003].

Teorema 5.2 (Eliminação de igualdades em semi-anéis). *Se S é um semi-anel que satisfaz a propriedade de finitude, então para todo par de matrizes $A, B \in S^{n \times p}$ o semimódulo $X = \{x \in S^p \mid Ax = Bx\}$ é finitamente gerado, ou seja, X é gerado em um número finito de passos.*

No caso particular do \mathbb{R}_{\max} o número mínimo de geradores de um hiperplano de \mathbb{R}_{\max}^q é da ordem q^2 [Katz, 2003]. Como consequência disso, o algoritmo que se obtém a partir do Teorema 5.2 para calcular um conjunto de geradores possui complexidade exponencial [Katz, 2003]. Entretanto, é possível incorporar a este algoritmo a construção de famílias geradoras mínimas e técnicas de eliminação que reduzem o tempo de execução. Para busca de ao menos uma solução com componentes finitas do sistema $Ax = Bx$ existem três métodos que podem ser utilizados: a técnica de Borriello e Walkup [1998], o algoritmo de Cuninghame-Green e Butkovic [2003] e a redução do problema a busca de um subponto fixo de uma função $\min - \max$ [Gaubert e Gunawardena, 1998]. Estes métodos são brevemente apresentados em Katz [2003] e são importantes para calcular um semimódulo $S^\dagger \subset S$ geometricamente (A, B) -invariante.

5.3.2 Semimódulos (geometricamente) (A, B) -Invariantes

Uma especificação para um sistema modelado através da equação (5.14) pode ser dada por um semimódulo $\mathcal{K} \subset S^n$, e deseja-se computar o conjunto máximo de estados iniciais \mathcal{K}^\dagger para o qual exista uma seqüência de vetores de controle que façam o estado do sistema (equação (5.14)) permanecer em \mathcal{K} para sempre, ou seja, $x(k) \in \mathcal{K}$ para todo $k \geq 0$. Para tratar este problema é necessário expressar a definição e o lema a seguir Katz [2007]:

Definição 5.19 (Semimódulo algebricamente (A, B) -invariante). *Dadas as matrizes $A \in S^{n \times n}$ e $B \in S^{n \times q}$, um semimódulo $X \subset S^n$ é geometricamente (A, B) -invariante se, para todo $x \in X$, existe um $u \in S^q$ tal que $Ax \oplus Bu$ permaneça em X .*

Lema 5.3. *Se $\mathcal{K} \subset S^n$ é um semimódulo, então \mathcal{K}^\dagger é o máximo semimódulo geometricamente (A, B) -invariante contido em \mathcal{K} .*

A partir da prova do Lema 5.3 (Katz [2003]) é possível observar que todo semimódulo (geometricamente) (A,B) -invariante contido em \mathcal{K} está também contido em \mathcal{K}^\uparrow , logo é possível concluir que \mathcal{K}^\uparrow é o maior semimódulo (A,B) -invariante contido em \mathcal{K} . Então, o problema pode ser descrito da seguinte maneira: dado um semimódulo $\mathcal{K} \in \mathbb{Z}_{\max}^n$, calcular o máximo semimódulo geometricamente (A,B) -invariante \mathcal{K}^\uparrow contido em \mathcal{K} .

Em Katz [2003] o algoritmo clássico de ponto fixo que é utilizado para calcular o máximo subespaço (A,B) -invariante contido em um dado espaço, apresentado em Wonham [1985], é generalizado para o semi-anel max-plus. Define-se $\mathcal{B} = \text{Im } B$ e considera-se o *self-map* ϕ do conjunto de subsemimódulos de \mathcal{S}^n , dado por:

$$\phi(\mathcal{X}) = \mathcal{X} \cap A^{-1}(\mathcal{X} \ominus \mathcal{B}) \quad (5.15)$$

onde $A^{-1}(\mathcal{Y}) = \{u \in \mathcal{S}^n | Au \in \mathcal{Y}\}$ e $\mathcal{Z} \ominus \mathcal{Y} = \{u \in \mathcal{S}^n | \exists y \in \mathcal{Y}, u \oplus y \in \mathcal{Z}\}$ para todo $\mathcal{Z}, \mathcal{Y} \subset \mathcal{S}^n$.

Dados os conjuntos de geradores dos semimódulos \mathcal{Z} e \mathcal{Y} finitamente gerados, os semimódulos $\mathcal{Y} \ominus \mathcal{Z}$, $A^{-1}(\mathcal{Y})$ e $\mathcal{Y} \cap \mathcal{Z}$ podem ser expressos pela imagem das matrizes que representam os conjuntos de soluções de sistemas max-plus lineares na forma $Dx = Cx$. Estes conjuntos podem ser computados utilizando métodos de eliminação de Butkovic e Hegedüs [1984] e Gaubert [1992]. Então, quando \mathcal{X} é finitamente gerado, o conjunto de geradores de $\phi(\mathcal{X})$ pode ser calculado através dos mesmos algoritmos.

Lema 5.4. *Um semimódulo $\mathcal{X} \subset \mathcal{S}^n$ é (geometricamente) (A,B) -invariante se e somente se $\mathcal{X} = \phi(\mathcal{X})$.*

É possível definir a seguinte seqüência de semimódulos:

$$\begin{aligned} \mathcal{X}_1 &= \mathcal{K} \\ \mathcal{X}_{r+1} &= \phi(\mathcal{X}_r), \forall r \in \mathbb{N}. \end{aligned} \quad (5.16)$$

E obtém-se o seguinte lema [Katz, 2003]:

Lema 5.5. *Seja $\mathcal{K} \subset \mathcal{S}^n$ um semimódulo arbitrário. A seqüência de semimódulos $\{\mathcal{X}_r\}_{r \in \mathbb{N}}$ definida na equação (5.16) é decrescente, isto é, $\mathcal{X}_{r+1} \subset \mathcal{X}_r$ para todo $r \in \mathbb{N}$. Além disso, se é definido $\mathcal{X}_w = \bigcap_{r \in \mathbb{N}} \mathcal{X}_r$ então cada semimódulo geometricamente (A,B) -invariante contido em \mathcal{K} está também contido em \mathcal{X}_w . Em particular, isto diz que $\mathcal{K}^\uparrow \subset \mathcal{X}_w$.*

É possível especificar condições suficientes para o semimódulo \mathcal{K} quando $\mathcal{S} = \mathbb{Z}_{\max}$, a fim de garantir que a seqüência definida na equação (5.16) se estabilize. Para isso é necessário introduzir a definição de volume de um semimódulo de \mathbb{Z}_{\max}^n [Katz, 2007].

Definição 5.20 (Volume). Seja $\mathcal{K} \subset \mathbb{Z}_{\max}^n$ um semimódulo. Chama-se volume de \mathcal{K} ($\text{vol}(\mathcal{K})$), a cardinalidade do conjunto $\{x \in \mathcal{K} \mid x_1 \oplus \cdots \oplus x_n = 0\}$, isto é, $\text{vol}(\mathcal{K}) = \text{card}(\{x \in \mathcal{K} \mid x_1 \oplus \cdots \oplus x_n = 0\})$. Além disso, se $K \in \mathbb{Z}_{\max}^{n \times p}$, representa-se por $\text{vol}(K)$ o volume do semimódulo $\mathcal{K} = \text{Im } K$, ou seja, $\text{vol}(\mathcal{K}) = \text{vol}(\text{Im } K)$.

Após definir o conceito de volume é possível expressar o seguinte lema:

Lema 5.6. Sejam $A \in \mathbb{Z}_{\max}^{n \times n}$, $B \in \mathbb{Z}_{\max}^{n \times p}$ e $C \in \mathbb{Z}_{\max}^{p \times q}$ matrizes e $\mathcal{Z}, \mathcal{Y} \subset \mathbb{Z}_{\max}^n$ semimódulos. Então as seguintes propriedades são verdade:

1. $\mathcal{Y} \subset \mathcal{Z} \Rightarrow \text{vol}(\mathcal{Y}) \subset \text{vol}(\mathcal{Z})$,
2. Se $\text{vol}(\mathcal{Y}) < \infty$, então $\mathcal{Y} \subsetneq \mathcal{Z} \Rightarrow \text{vol}(\mathcal{Y}) < \text{vol}(\mathcal{Z})$,
3. $\text{vol}(A\mathcal{Y}) \leq \text{vol}(A)$ e então $\text{vol}(AB) \leq \text{vol}(A)$,
4. $\text{vol}(A\mathcal{Y}) \leq \text{vol}(\mathcal{Y})$ e então $\text{vol}(AB) \leq \text{vol}(B)$,
5. $\text{vol}(ABC) \leq \text{vol}(B)$,
6. Se $P \in \mathbb{Z}_{\max}^{n \times n}$ e $Q \in \mathbb{Z}_{\max}^{p \times p}$ são inversíveis então $\text{vol}(PBQ) = \text{vol}(B)$,
7. $\text{vol}(A) = \text{vol}(A^T)$.

As provas de todas as propriedades estão descritas em [Katz, 2003].

O próximo teorema proporciona uma condição sobre a especificação \mathcal{K} que garante que a sucessão de semimódulos $\{\mathcal{X}_r\}_{r \in \mathbb{N}}$ se estabiliza em um número finito de passos.

Teorema 5.7. Seja $\mathcal{K} \subset \mathbb{Z}_{\max}^n$ um semimódulo com volume finito. Então, para todo $A \in \mathbb{Z}_{\max}^{n \times n}$ e $B \in \mathbb{Z}_{\max}^{n \times q}$, o máximo semimódulo (geometricamente) (A, B) -invariante \mathcal{K}^\uparrow contido em \mathcal{K} é finitamente gerado. Além disso, se a sequência de semimódulos é definida como descrito na equação (5.16) então $\mathcal{K}^\uparrow = \mathcal{X}_k$ para algum $k \leq \text{vol}(\mathcal{K}) + 1$.

Este teorema é muito útil para resolver problemas onde as especificações tem volume finito. Isto ocorre com frequência nos problemas onde \mathcal{K} modela condições de estabilidade, como requisitos de não ultrapassar um limite de tempo. Assumindo que um sistema dado pela equação (5.14) é representado por um grafo a eventos temporizado, um caso típico de um semimódulo \mathcal{K} é dado por:

$$\mathcal{K} = \{x \in \mathbb{Z}_{\max}^n \mid x_i - x_j \leq d_{ij}, \quad \forall i, j; \quad 1 \leq i, j \leq n\} \quad (5.17)$$

onde $D = (d_{ij})$ é uma matriz com entradas em $\mathbb{Z} \cup \{+\infty\}$. O vetor de estados $x(k)$ do sistema pertence a \mathcal{K} se e somente se $x(k)_i - x(k)_j \leq d_{ij}$ para todo $1 \leq i, j \leq n$, o que significa que o tempo que transcorre entre o k -ésimo disparo da transição j e o k -ésimo disparo da transição i deve ser menor ou igual d_{ij} .

Na prática pode-se assumir que todas as entradas de D são finitas, trocando $+\infty$ por uma constante de valor bastante alto. Neste caso é possível mostrar que \mathcal{K} definido pela equação (5.17) tem volume finito [Katz, 2007].

O semimódulo dado pela equação (5.17) pode ser expresso como $\mathcal{K} = \{x \in \mathbb{Z}_{\max}^n \mid Ex \leq x\}$, onde $E = (-D)^T$ e o seguinte lema pode ser definido:

Lema 5.8. *Se a matriz E é irredutível, então o semimódulo $\mathcal{K} = \{x \in \mathbb{Z}_{\max}^n \mid Ex \leq x\}$ tem volume finito. Além disso, se E tem raio spectral maior que a unidade (isto é 0) então \mathcal{K} é reduzido ao vetor nulo.*

Se E é uma matriz irredutível e o raio spectral menor que a unidade, sabe-se que $\mathcal{K} = \text{Im}E^*$, onde $E^* = I \oplus E \oplus \dots \oplus E^{n-1}$, sendo n é a ordem da matriz.

5.3.3 Semimódulos (algebricamente) (A,B)-Invariantes

Após encontrar o máximo semimódulo \mathcal{K}^\uparrow geometricamente (A,B)-invariante contido em um semimódulo \mathcal{K} o problema passa a ser determinar se existe uma realimentação linear $u(k) = Fx(k-1)$ de maneira que o sistema autônomo resultante

$$x(k) = (A \oplus BF)x(k-1) \quad (5.18)$$

mantenha \mathcal{K}^\uparrow invariante, ou seja, que qualquer trajetória de um sistema dado pela equação (5.18) esteja completamente contida em \mathcal{K}^\uparrow quando o seu estado inicial estiver contido em \mathcal{K}^\uparrow . Se existir uma realimentação linear diz-se que \mathcal{K}^\uparrow é um semimódulo algebricamente (A,B)-invariante. Do ponto de vista dinâmico, isto significa que as trajetórias do sistema dado pela equação (5.18) com estado inicial contido em \mathcal{K}^\uparrow permanecerão para sempre em \mathcal{K}^\uparrow através de uma escolha de controle apropriada.

Definição 5.21. *Dadas as matrizes $A \in \mathbb{Z}_{\max}^{n \times n}$ e $B \in \mathbb{Z}_{\max}^{n \times q}$ diz-se que um semimódulo $\mathcal{X} \subset \mathbb{Z}_{\max}^n$ é algebricamente (A,B)-invariante se existe uma matriz $F \in \mathbb{Z}_{\max}^{q \times n}$ tal que $(A \oplus BF)\mathcal{X} \subset \mathcal{X}$.*

Todo semimódulo algebricamente (A,B)-invariante é também geometricamente (A,B)-invariante. Entretanto, a recíproca nem sempre é verdadeira, resultando numa diferença quanto à teoria clássica que diz que sempre é possível encontrar uma realimentação linear ao encontrar um subespaço (A,B)-invariante [Katz, 2003].

Sendo \mathcal{X} um semimódulo geometricamente (A,B)-invariante e finitamente gerado sabe-se que existe um $r \in \mathbb{N}$ e uma matriz $H \in \mathbb{Z}_{\max}^{n \times r}$ tal que $\mathcal{X} = \text{Im}H$. Então, a partir da Definição 5.21 sabe-se que \mathcal{X} é algebricamente (A,B)-invariante se e somente se existem as matrizes $F \in \mathbb{Z}_{\max}^{q \times n}$ e $G \in \mathbb{Z}_{\max}^{r \times r}$ tal que $(A \oplus BF)H = HG$. Como a equação é um sistema de equações max-plus linear não homogêneo, o conjunto das soluções (F, G) pode ser calculado utilizando métodos gerais de eliminação [Gaubert,

1992]. Entretanto, para afirmar que $\mathcal{X} = \text{Im}H$ é algebricamente (A,B)-invariante é suficiente saber se o sistema de equações $(A \oplus BF)H = HG$ tem ao menos uma solução [Katz, 2007], e existem métodos alternativos para tal [Cuninghame-Green e Butkovic, 2003].

O problema da existência de uma solução (com entradas finitas) de um sistema deste tipo pode ser reduzido ao problema da existência de um sub-ponto fixo de uma função min-max [Katz, 2007]. Neste casos deve-se introduzir uma nova variável t de forma a obter um novo sistema de equações que seja homogêneo $(At \oplus BF)H = HG$. Utilizando a teoria da residuação tem-se que (t, F, G) é uma solução do sistema se e somente se verifica que:

$$\begin{aligned} t &\leq (AH) \bowtie (HG), \\ F &\leq B \bowtie (HG) \not\bowtie H, \\ G &\leq H \bowtie ((At \oplus BF)H). \end{aligned} \tag{5.19}$$

onde $D \bowtie C$ é definido como $\sup\{E \in \mathbb{Z}_{\max}^{p \times r} | DE \leq C\}$ para todo $D \in \mathbb{Z}_{\max}^{n \times r}$ e $C \in \mathbb{Z}_{\max}^{n \times r}$ (a função $\not\bowtie$ é definida de maneira análoga). Se é possível encontrar um subponto fixo da função min-max definida pelo lado direito de 5.19, então (t, F, G) é uma solução para $(At \oplus BF)H = HG$ com $t \neq -\infty$ e $t^{-1}F = (-t)F$ é a realimentação procurada.

5.3.4 Exemplo

Para entender melhor os conceitos relacionados aos semimódulos (A,B)-invariantes um exemplo extraído de [Katz, 2003] é apresentado. As matrizes A e B que descrevem um sistema são:

$$A = \begin{pmatrix} -\infty & 0 & -\infty \\ -\infty & -\infty & 0 \\ 0 & -\infty & -\infty \end{pmatrix} \quad \text{e} \quad B = \begin{pmatrix} 2 & 0 \\ 0 & 2 \\ 0 & 0 \end{pmatrix}$$

Uma especificação para o sistema é dada por um semimódulo $\mathcal{K} = \text{Im}(K)$, sendo:

$$K = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}$$

Primeiramente se calcula o máximo semimódulo \mathcal{K}^\uparrow geometricamente (A,B)-invariante, para então verificar se é possível obter uma realimentação linear para ele. Para encontrar \mathcal{K}^\uparrow é necessário calcular a sucessão de semimódulos $\{\mathcal{X}\}_{r \in \mathbb{N}}$ definida na equação (5.16). Sendo assim é possível escrever

$$\begin{aligned}\mathcal{X}_1 &= \mathcal{K} \\ \mathcal{X}_2 &= \varphi(\mathcal{X}_1)\end{aligned}$$

onde:

$$\varphi(\mathcal{X}_1) = \mathcal{X}_1 \cap A^{-1}(\mathcal{X}_1 \ominus \mathcal{B}).$$

Para encontrar uma matriz que represente uma família geradora do semimódulo \mathcal{X}_1 é possível utilizar o Maxplus Toolbox disponível no Scilab [Max Plus, 1998]. Através da função denominada `mpsolve` é possível resolver sistemas lineares max-plus da forma $Dx = Cx$. Para manipular as funções disponíveis no toolbox são utilizadas as matrizes que representam as famílias geradoras dos semimódulos $\mathcal{B} = \text{Im}(\mathcal{B})$ e $\mathcal{X}_1 = \text{Im}(\mathcal{X}_1) = \text{Im}(K)$

A matriz X_2 , tal que $\mathcal{X}_2 = \text{Im}(X_2)$, calculada é:

$$X_2 = \begin{pmatrix} 3 & 6 & 4 \\ 4 & 5 & 3 \\ 3 & 5 & 5 \end{pmatrix}$$

Se a imagem dos semimódulos \mathcal{X}_1 e \mathcal{X}_2 são iguais então $\mathcal{K}^\dagger = \mathcal{X}_2$. O teste das imagens é realizado através das funções `includespan` e `equalspan` do Maxplus Toolbox do Scilab. Para o caso acima $\text{Im}(X_2) \subset \text{Im}(X_1)$ mas $\text{Im}(X_1) \neq \text{Im}(X_2)$, portanto é necessário calcular \mathcal{X}_3 :

$$\mathcal{X}_3 = \varphi(\mathcal{X}_2) = \mathcal{X}_2 \cap A^{-1}(\mathcal{X}_2 \ominus \mathcal{B})$$

onde a família geradora de \mathcal{X}_3 é dada pela matriz:

$$X_3 = \begin{pmatrix} 19 & 17 & 16 \\ 18 & 16 & 17 \\ 18 & 18 & 16 \end{pmatrix}$$

Através do teste das imagens conclui-se que $\text{Im}(X_3) = \text{Im}(X_2)$, e portanto $\mathcal{K}^\dagger = \mathcal{X}_3 = \text{Im}(X_3)$. Sabe-se *a priori* que o $\varphi(\mathcal{X})$ vai estabilizar pois K é uma matriz composta apenas de componentes finitos, logo ela é irredutível e portanto tem volume finito, garantindo a estabilização (Teorema 5.7).

Após calcular \mathcal{K}^\dagger deseja-se saber se ele é algebricamente (A,B)-invariante. O semimódulo $\mathcal{K}^\dagger = \text{Im}(X_3)$ é algebricamente (A,B)-invariante se e somente se o sistema max-plus linear homogêneo

$$(At \oplus BF)H = HG$$

admite uma solução (t, F, G) com $t \neq -\infty$, sendo $H = X_3 \otimes (-16)$. Para o caso do exemplo desenvolvido o sistema é dado por:

$$\left[\begin{pmatrix} -\infty & 0 & -\infty \\ -\infty & -\infty & 0 \\ 0 & -\infty & -\infty \end{pmatrix} t \oplus \begin{pmatrix} 2 & 0 \\ 0 & 2 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \end{pmatrix} \right] \begin{pmatrix} 3 & 1 & 0 \\ 2 & 0 & 1 \\ 2 & 2 & 0 \end{pmatrix} =$$

$$\begin{pmatrix} 3 & 1 & 0 \\ 2 & 0 & 1 \\ 2 & 2 & 0 \end{pmatrix} \begin{pmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{pmatrix}$$

Como apresentado anteriormente, não é necessário encontrar todas as soluções do sistema, apenas uma com t não nulo. A busca de uma solução para equação (5.19) também pode ser implementada através de funções do Maxplus Toolbox pois os operadores \backslash e $/$ estão implementados. É necessário definir valores iniciais para t, F , e G e então calcular sucessivamente as inequações até convergir para um resultado. Quando isto ocorre encontra-se uma solução para (t, F, G) .

Sendo os valores iniciais de t, F e G :

$$t_0 = 0, \quad F_0 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad \text{e} \quad G_0 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

são encontradas duas matrizes F que garantem a (A,B)-invariância de \mathcal{K}^\uparrow . São elas:

$$F_1 = \begin{pmatrix} -1 & -2 & -2 \\ -1 & -2 & -2 \end{pmatrix} \quad F_2 = \begin{pmatrix} -1 & -2 & -2 \\ -2 & -2 & -2 \end{pmatrix}$$

Ou seja, ambos os testes $(A \oplus BF_1)\mathcal{K}^\uparrow \subset \mathcal{K}^\uparrow$ e $(A \oplus BF_2)\mathcal{K}^\uparrow \subset \mathcal{K}^\uparrow$ são verdadeiros.

5.4 Conclusão

Ao longo deste capítulo alguns aspectos relacionados a teoria de dióides foram apresentados, sendo explorado um exemplo ilustrativo e conceitos teóricos. Além disso, o formalismo que vai ser adotado na modelagem discreta dos sistemas de tráfego, a álgebra max-plus, foi estudado.

A coordenação semafórica em um sistema de tráfego urbano é modelada através de uma abordagem puramente discreta no Capítulo 6, onde o problema de coordenação semafórica para um sistema composto de arteriais é discutido. A modelagem do sistema é baseada na álgebra max-plus e o problema de síntese de controle é resolvido utilizando os conceitos sobre semimódulos (A,B)-invariantes apresentados neste capítulo.

Capítulo 6

Modelagem e Controle de Sistemas de Tráfego através da Álgebra Max-Plus

Um dos objetivos desta tese é resolver o problema de modelagem de sistemas de tráfego urbano e síntese de controle semafórico sob o enfoque dos sistemas a eventos discretos (SEDs), e o formalismo adotado é a álgebra max-plus [Baccelli et al., 1992]. O trabalho propõe uma nova abordagem do problema de controle semafórico, enfocando a coordenação semafórica de arteriais. A solução proposta utiliza o conceito de semimódulos (A,B)-invariantes [Katz, 2007] para obter um sistema controlado com realimentação de estado.

A próxima seção do capítulo apresenta alguns trabalhos relacionados com os sistemas de tráfego e SEDs, os quais inspiraram a solução proposta para o problema de coordenação semafórica. A proposta desenvolvida neste capítulo é apresentada na seção 6.2. Aspectos relativos à implementação dos modelos obtidos e resultados de simulação também são apresentados ao longo do capítulo.

6.1 Trabalhos Relacionados

Os sistemas de tráfego são sistemas sujeitos a restrições de sincronização, e podem ser modelados através da álgebra max-plus [Baccelli et al., 1992]. Existem na literatura alguns trabalhos que utilizam este formalismo para resolver problemas de coordenação em sistemas viários e ferroviários.

Em van Egmond e Olsder [1998] uma proposta de modelagem e análise baseada nos autovalores e autovetores do sistema max-plus linear é utilizada para calcular planos semafóricos. Dado um conjunto de rotas de tráfego o problema que os autores desejam resolver é o de determinar planos semafóricos que minimizam o tempo de espera dos veículos pelo sinal verde. Assume-se que cada veículo segue uma rota pré-determinada e ao longo desta rota não são permitidas escolhas. Os autores buscam a minimização do tempo de espera através da sincronização dos semáforos.

Nesta abordagem cada interseção tem o seu próprio plano semafórico, que depende do tipo de interseção (cruzamento normal ou T-cruzamento). Considera-se que o início de verde em cada estágio

caracteriza um evento, e quando este evento ocorre todos os veículos que estavam esperando se movem para o próximo cruzamento. O conjunto de veículos que se movem para uma mesma interseção formam um pelotão. O tamanho do pelotão é ignorado, considera-se um ponto indivisível, e como consequência, apenas o início do verde em cada estágio é sincronizado. A idéia utilizada para definir equações max-plus para o sistema é: *não deixe os veículos esperando no semáforo, mas os semáforos esperando os veículos* [van Egmond e Olsder, 1998].

Para definir as equações max-plus do sistema de tráfego é necessário conhecer os movimentos permitidos nos estágios. A Figura 6.1 mostra os movimentos permitidos para os estágios definidos para um cruzamento normal e para um T-cruzamento.

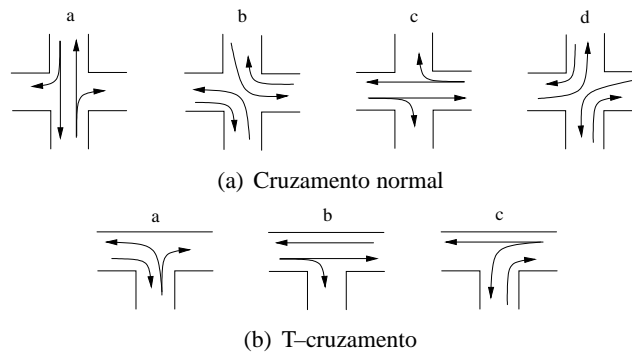


Figura 6.1: Estágios para um cruzamento normal e um T-cruzamento

Um evento é caracterizado pela abertura do sinal verde em um estágio, e $x_i(k)$ corresponde ao tempo da k -ésima abertura do sinal verde para o estágio. As equações max-plus para o sistema são geradas seguindo as seguintes regras [van Egmond e Olsder, 1998]:

1. Um estágio deve permanecer em verde até que todos os pelotões cruzando uma via tenham chegado.
2. O tempo de duração entre dois estágios sucessivos (na mesma interseção) deve ter um valor mínimo.

Considere o sistema da Figura 6.2. As linhas indicam as rotas pré-determinadas e o tempo de viagem entre duas interseções é 36 segundos.

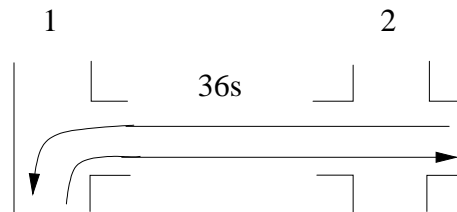


Figura 6.2: Pequeno exemplo

Os estágios considerados são os definidos na Figura 6.1 (os estágios da Figura 6.1(b) são rotacionados 90°). A ocorrência dos estágios da interseção 1 seguem a seguinte ordem: $x_{1a} - x_{1b} - x_{1c} -$

$-x_{1d}$; para a interseção 2 a ordem é $x_{2a} - x_{2b} - x_{2c}$. Se o tempo mínimo de verde é 10 segundos, é possível escrever as seguintes equações max-plus lineares para o exemplo:

$$x_{1a}(k+1) = (10 \otimes x_{1c}(k)) \oplus (36 \otimes x_{2c}(k)) = 10 x_{1c} \oplus 36 x_{2c}$$

$$x_{1b}(k+1) = 10 x_{1a}(k+1)$$

$$x_{1c}(k+1) = 10 x_{1b}(k+1)$$

$$x_{2a}(k+1) = 10 x_{2d}(k)$$

$$x_{2b}(k+1) = 10 x_{2a}(k+1)$$

$$x_{2c}(k+1) = 10 x_{2b}(k+1) \oplus 36 x_{1a}(k)$$

$$x_{2d}(k+1) = 10 x_{2c}(k+1)$$

onde os índices $1a, 1b, \dots, 2d$ correspondem aos diferentes estágios das interseções 1 e 2. A Figura 6.3 apresenta o grafo de eventos (GET) correspondente as equações definidas para o exemplo. Cada transição representa um estágio e os lugares representam os tempo mínimos de verde de cada estágio ou os tempos de deslocamento.

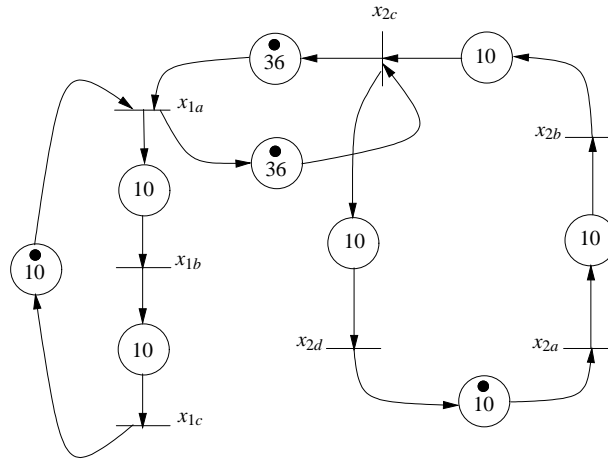


Figura 6.3: GET para pequeno exemplo

Para o caso do exemplo os estágios que precisam ser sincronizados são os representados pelas transições x_{1a} e x_{2c} , ou seja, os movimentos permitidos no estágio x_{1a} só tem permissão de passagem quando o pelotão que partiu da interseção 2 já tenha alcançado a interseção 1. O mesmo raciocínio é aplicado ao estágio x_{2c} .

O objetivo do trabalho proposto por van Egmond e Olsder [1998] é definir planos semafóricos de tempo fixo para o sistema. Sabe-se que o início de verde em cada estágio se repete a cada λ unidades de tempo, para algum $\lambda \in \mathbb{R}$. O tempo λ corresponde ao tempo de ciclo T , onde T corresponde ao tempo necessário para que todos os estágios sejam contemplados.

Segundo van Egmond e Olsder [1998] o plano semafórico em tempo fixo pode ser escrito como:

$$x(k+1) = \lambda x(k), \quad k = 0, 1, \dots$$

para período igual a 1, caso contrário cada estágio se repete em média a cada λ unidades de tempo.

Sabe-se que $x(k+1) = Ax(k)$, sendo possível encontrar um vetor v de modo que:

$$Av = \lambda v$$

onde λ e v são os autovalores e autovetores de A . Neste caso, o autovalor λ determina, então, o tempo de ciclo de todo o sistema composto pelas interseções [van Egmond e Olsder, 1998]. Se A é irredutível, então existe um autovalor único que é igual ao máximo de todos ciclos médios dos circuitos do grafo de precedência de A . Para o caso do exemplo aqui descrito o tempo de ciclo é $\lambda = 40$ segundos.

O trabalho de Vries et al. [1998] busca sincronizar um sistema de trens. A álgebra max-plus é utilizada para definir a tabela de horários para as partidas dos trens a fim de permitir conexões. Entretanto, um trem somente espera outro trem por um período determinado de tempo, caso contrário a conexão é quebrada. Isto é realizado com o intuito de evitar a propagação de atrasos através da rede.

Os exemplos apresentados em de Vries et al. [1998] são semelhantes ao desenvolvido na seção 5.1, onde um sistema max-plus linear é apresentado através de uma pequena rede de trens. A Figura 6.4 apresenta o exemplo utilizado no trabalho de Vries et al. [1998]. Na rede da Figura 6.4 existe uma linha de trem de P para S via Q e vice-versa. Além disso, existe uma linha de ida e volta de Q para R . A conexão dos trens partindo de P e S com os que partem para R é dada na estação Q .

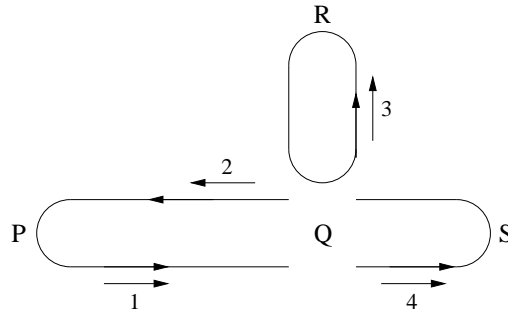


Figura 6.4: Exemplo – rede de trens

Seja $x_i(k)$ ($i = 1, 2, 3, 4$) o tempo da k -ésima partida de trem na direção i . Um trem só pode partir na direção i quando três condições são satisfeitas:

1. O trem deve ter chegado na estação. Se um trem vindo na direção j vai continuar na direção i ele deve obedecer a seguinte condição:

$$x_i(k+1) \geq a_{ij} \otimes x_j(k)$$

onde a_{ij} é o tempo de viagem entre j e i .

2. A conexão com outros trens deve ser possível. Isto leva à seguinte condição:

$$x_i(k+1) \geq a_{il} \otimes x_l(k)$$

onde l indica as direções de onde partem trens que podem realizar conexão na direção i .

3. O trem não pode partir antes do instante d_i previsto na tabela de horários, ou seja, a seguinte condição deve ser respeitada:

$$x_i(k+1) \geq d_i.$$

Se um trem parte imediatamente após todas as condições serem satisfeitas, o instante de partida é dado pelas seguintes equações:

$$x_i(k+1) = a_{ij}x_j(k) \oplus d_i(k+1) \oplus \bigoplus_l (a_{il}(k) x_l(k)),$$

$$x(k+1) = Ax(k) \oplus d(k+1).$$

O modelo max-plus para o sistema da Figura 6.4, desconsiderando a tabela de horários, é dado pelas seguintes equações:

$$\begin{aligned} x_1(k+1) &= a_2(k) x_2(k) \\ x_2(k+1) &= a_3(k)x_3(k) \oplus a_4(k)x_4(k) \\ x_3(k+1) &= a_1(k)x_1(k) \oplus a_3(k)x_3(k) \oplus a_4(k)x_4(k) \\ x_4(k+1) &= a_1(k)x_1(k) \oplus a_3(k)x_3(k) \end{aligned}$$

Quando os tempos de viagem $a_i(k)$ são determinísticos e invariantes no tempo, o comportamento do sistema $x(k+1) = Ax(k)$ é determinado pelo autovalor λ de A . Isto significa que para uma escolha apropriada de $x(1)$ a cada λ unidades de tempo um trem pode partir. A inclusão de uma tabela de horários no modelo deve garantir que duas partidas consecutivas na mesma direção só podem acontecer num intervalo de ao menos λ unidades de tempo, caso contrário o sistema não poderá operar sobre a tabela de horários definida.

Seja $d(1)$ um vetor que contém os instantes de partida do primeiro trem em cada direção. Para que o escalonamento seja o mais regular possível a tabela de horários deve satisfazer a seguinte condição:

$$d(k+1) = \tau \otimes d(k) \quad k = 1, 2, 3, \dots$$

onde τ representa o período da tabela de horários. O período τ deve satisfazer a condição $\tau \geq \lambda$ onde λ é o autovalor da matriz A . Além da escolha apropriada do período, a escolha do vetor $d(1)$ também tem influência na propagação dos atrasos.

Em de Vries et al. [1998], os autores não querem que um trem espere muito tempo por uma conexão. O trem na direção i parte no instante $d_i(k+1) + M_{il}$, com $M_{il} > 0$, caso o trem vindo na direção l para conexão não tenha chegado ainda. A quantidade M_{il} pode ser interpretada como o máximo tempo que o trem na direção i pode esperar por um trem vindo na direção l após o seu horário previsto para partir. O modelo max-plus passa a ser o seguinte:

$$x_i(k+1) = a_{ij}x_j(k) \oplus d_i(k+1) \oplus \bigoplus_l (a_{il}(k) x_l(k) \oplus' M_{il} d_i(k+1))$$

onde $a \oplus' b = \min(a, b)$. A idéia desta abordagem é que o $(k+1)$ -ésimo trem na direção i sempre espera o k -ésimo trem vindo na direção l até $d_i(k+1) + M_{il}$, mesmo quando o atraso do trem vindo na direção l é maior que esta quantidade. Como consequência é possível que um trem na direção i espere desnecessariamente um trem que não vai chegar.

Uma solução para evitar esperas desnecessárias e também minimizar a propagação dos atrasos na rede é tornar as conexões controláveis. A introdução da variável $u_{il}(k)$ indica se o k -ésimo trem na direção i vai esperar por uma conexão com o trem vindo na direção l . A introdução da variável u_{il} leva ao seguinte modelo:

$$x_i(k+1) = a_{ij}x_j(k) \oplus d_i(k+1) \oplus \bigoplus_l (a_{il}(k) x_l(k) \oplus' M_{il} d_i(k+1)) u_{il}$$

onde u_{il} é dado por

$$u_{il} = \begin{cases} 0, & \text{quando } i \text{ espera por } l \\ \varepsilon, & \text{caso contrário.} \end{cases}$$

Crerios para decidir quando um trem vai esperar uma conexão, ou seja, para definir o valor de u_{il} são discutidos em de Vries et al. [1998]. É possível citar que o número de passageiros que irão realizar a conexão, a possibilidade de atender o número máximo de conexões, dentre outros, são alguns dos pontos analisados.

Em Katz [2007] o mesmo exemplo da Figura 6.4 é apresentado. Entretanto, neste trabalho o problema de sincronização dos trens é resolvido através do uso de semimódulos (A,B)-invariantes. As condições que devem ser respeitadas são exatamente as mesmas apresentadas no trabalho de de Vries et al. [1998]. A diferença é que agora a tabela de horário é considerada uma entrada de controle, e o sistema é analisado segundo a equação $x(k+1) = Ax(k) \oplus Bu(k+1)$. A condição que afirma que um trem não pode partir antes do instante previsto na tabela de horários agora é definida como $x_i(k+1) \geq u_i(k+1)$.

O trabalho de Katz [2007] mostra como calcular uma tabela de horários para o sistema de modo que as condições apresentadas sejam sempre respeitadas. Especificações que definem limites para os intervalos entre partidas dos trens e para o tempo de espera dos passageiros são definidas como semimódulo $\mathcal{K} = \{x \in \mathbb{Z}_{\max}^n \mid x_i - x_j \leq d_{ij}, \forall i, j; 1 \leq i, j \leq n\}$ (equação (5.17)). O problema então passa a ser determinar se existe um semimódulo (A,B)-invariante \mathcal{K}^\uparrow contido em \mathcal{K} e então calcular um controle com realimentação de estados que garanta que as restrições sejam sempre respeitadas. O método utilizado para encontrar \mathcal{K}^\uparrow foi apresentado na seção 5.3 do Capítulo 5.

Os trabalhos aqui apresentados serviram de inspiração para a proposta de modelagem e controle que é apresentada a seguir.

6.2 Proposta para Modelagem e Controle de Sistemas de Tráfego

A álgebra max-plus é um formalismo eficiente para resolver o problema de controle de tráfego urbano quando se deseja realizar sincronização de semáforos. O modelo que é proposto nesta seção visa modelar as vias arteriais de um sistema viário. Uma via é chamada arterial quando possui uma extensão larga, a sua velocidade de percurso é média ou alta e o fluxo veicular intenso. As arteriais tem prioridade mais alta nos cruzamentos, ou seja, o tempo de verde da arterial é superior ao alocado para as vias secundárias ou transversais. Além disso, deseja-se uma sincronização dos semáforos que favoreça a progressão de pelotões ao longo de toda arterial.

A proposta aqui apresentada busca sincronizar os semáforos de uma malha viária, a partir dos tempos de verde nominais associados aos estágios. Além disso, o método proposto para modelagem define modelos específicos para as vias e os semáforos, garantindo uma descrição mais detalhada dos eventos que ocorrem no sistema.

As seguintes premissas devem ser consideradas para modelar um sistema de tráfego urbano:

- os veículos que se deslocam entre as interseções são agrupados em pelotões;
- um pelotão é considerado como um ponto indivisível. Seu tamanho é definido pelo tempo mínimo de verde do sistema controlado, ou seja, o tempo mínimo de verde tem que ser suficiente para permitir o cruzamento de um pelotão pela interseção;
- o conjunto mínimo de interseções é composto de duas unidades, como mostra a Figura 6.5. Os semáforos S_i estão relacionados com as interseções I_i com $i = 1, 2$, e d_{12} é o tempo de deslocamento entre I_1 e I_2 ;

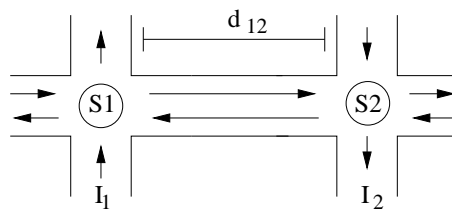


Figura 6.5: Sistema mínimo (2 interseções)

- os eventos que se deseja modelar são as trocas de sinal dos semáforos e a chegada dos pelotões nas interseções;
- a via arterial de mão dupla da Figura 6.5 pode ser modelada através de um grafo de evento (GET). A Figura 6.6 mostra o GET para a arterial, onde as transições x_{i1} e x_{i2} indicam a passagem de um pelotão pela interseção I_i com $i = 1, 2$ e conversões não são permitidas. Aos lugares que conectam as transições são associados os tempos $d_{12} = d_{21}$, que representam o tempo de deslocamento entre as duas interseções. A marcação dos lugares, que são nomeados d_{ij} , indicam a presença ou não de um pelotão na via;

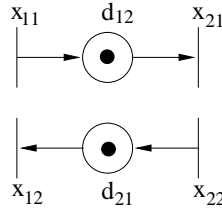


Figura 6.6: Modelo de uma via de mão dupla

- os semáforos podem estar com a indicação semafórica verde ou vermelha. O tempo de amarelo não é considerado isoladamente. O fato da indicação estar verde na direção da arterial significa que o sinal está vermelho na direção da via transversal, e vice-versa. O modelo utilizado para o semáforo está na Figura 6.7. As transições indicam a troca de sinal do semáforo, s_{i1} habilita o sinal verde e s_{i2} o sinal vermelho, os tempos r_i e g_i associados aos lugares representam, respectivamente, os tempos de vermelho e verde do semáforo i com $i = 1, 2$. Além disso, r_i e g_i também são utilizados para nomear os lugares. A marcação utilizada no modelo indica a habilitação do sinal verde ao inicializar a rede;

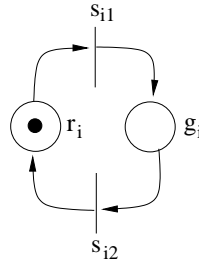


Figura 6.7: Modelo de um semáforo

- a conexão entre uma via e um semáforo se dá através de um lugar não temporizado entre as transições que habilitam o sinal verde para a arterial e as transições que habilitam a passagem de um pelotão pela via. A Figura 6.8 mostra a conexão entre uma via e um semáforo, onde um pelotão só pode ingressar na via $x_{11} \rightarrow x_{21}$ quando o semáforo s_1 estiver com o sinal verde, ou seja, quando a transição s_{11} for sensibilizada.

Após definir modelos isolados para uma via e um semáforo, e a conexão entre eles, é possível definir um GET que modele a malha viária da Figura 6.5. A malha viária é composta de duas interseções, e portanto é necessário modelar dois semáforos, um para cada interseção. Além disso, como a via é de mão dupla é necessário modelar a conexão para cada direção da via separadamente. O modelo da malha semaforizada está na Figura 6.9. Este modelo possui semáforos com temporização fixa pois os tempos de verde e vermelho são definidos *a priori* e não são alterados dinamicamente pelo GET, além disso não é previsto nenhum controle externo.

É possível introduzir entradas de controle no GET da Figura 6.9, como mostra a Figura 6.10. A introdução das transições u_i permite que os chaveamentos do sinal semafórico sejam atrasados em relação àqueles definidos pelos tempos mínimos de verde e vermelho, dados por g_i e r_i .

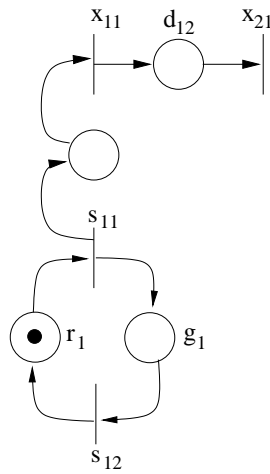


Figura 6.8: Conexão de uma via e um semáforo

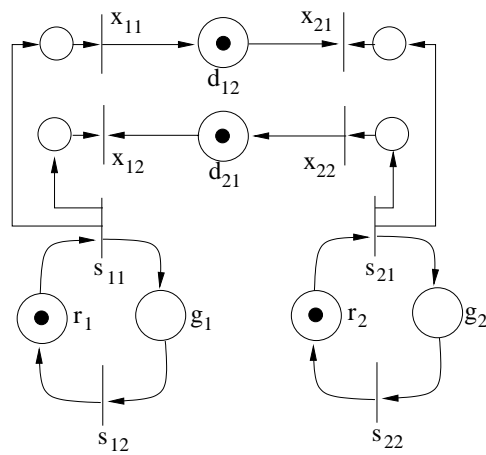


Figura 6.9: Modelo de uma malha semaforizada

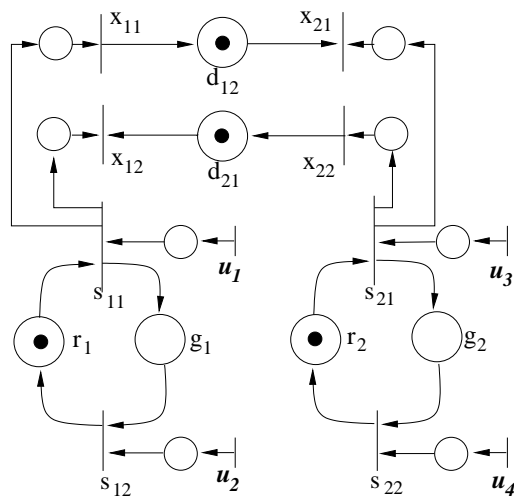


Figura 6.10: Modelo de uma malha com entradas de controle

O sistema da Figura 6.10 é um sistema max-plus linear definido pela equação $x(k) = Ax(k-1) \oplus Bu(k)$ (equação (5.14)) onde $A \in \mathbb{Z}_{\max}^{n \times n}$ e $B \in \mathbb{Z}_{\max}^{n \times q}$. As equações que representam o modelo da malha com entradas de controle são:

$$\begin{aligned} s_{11}(k) &= r_1 s_{12}(k-1) \oplus u_1(k) = \max(r_1 + s_{12}(k-1), u_1(k)) \\ s_{12}(k) &= g_1 s_{11}(k) \oplus u_2(k) \\ x_{11}(k) &= s_{11}(k) \\ x_{12}(k) &= d_{21} x_{22}(k-1) \oplus s_{11}(k) \\ s_{21}(k) &= r_2 s_{22}(k-1) \oplus u_3(k) \\ s_{22}(k) &= g_2 s_{21}(k) \oplus u_4(k) \\ x_{21}(k) &= d_{12} x_{11}(k-1) \oplus s_{21}(k) \\ x_{22}(k) &= s_{21}(k) \end{aligned}$$

onde as variáveis $s_{ij}(k)$, $x_{ij}(k)$ e $u_l(k)$, $i, j = 1, 2$ e $l = 1, \dots, 4$ representam o k -ésimo disparo das transições x_{ij} , s_{ij} e u_l da Figura 6.10.

As equações apresentadas modelam a sincronização da passagem dos pelotões pelos cruzamentos com a habilitação do sinal verde na direção da arterial. Entretanto, é necessário definir um conjunto de restrições operacionais para garantir o bom funcionamento do sistema controlado.

As restrições são definidas sobre as transições do GET, e visam impedir que uma determinada transição i ocorra antes de outra transição j acrescida de um limite de tempo w , ou seja, $i \geq j + w$. Restrições definidas desta forma podem ser escritas como um semimódulo \mathcal{K} e então o conceito de semimódulos (A,B)-invariantes proposto por Katz [2007] pode ser utilizado para calcular um controle realimentado para os sistemas de tráfego urbano.

Para o caso do controle de vias arteriais definem-se quatro conjuntos de restrições:

1. Um pelotão só pode cruzar uma interseção quando o semáforo estiver com sinal verde na direção da arterial. Para o modelo da Figura 6.10 são definidas as seguintes restrições:

$$\begin{aligned} s_{11}(k) - x_{1i}(k) &\leq 0 \\ s_{21}(k) - x_{2i}(k) &\leq 0 \quad i = 1, 2, \end{aligned}$$

ou seja, as transições $x_{1i}(k)$ e $x_{2i}(k)$ só podem ser sensibilizadas após a sensibilização de $s_{i1}(k)$. Se o resultado de $s_{11}(k) - x_{1i}(k)$ ou $s_{21}(k) - x_{2i}(k)$ é um valor negativo pode-se concluir que a abertura do sinal verde ocorreu antes da chegada de um pelotão. O resultado nulo é a situação ideal, onde o pelotão chega no cruzamento exatamente na abertura do sinal verde.

2. O tempo de verde deve ser longo o suficiente para absorver um pelotão. As seguintes restrições

podem ser escritas como:

$$\begin{aligned} x_{1i}(k) - s_{12}(k) &\leq -t_a \\ x_{2i}(k) - s_{22}(k) &\leq -t_a \quad i = 1, 2, \end{aligned}$$

onde t_a é o tempo de absorção. O valor utilizado nesta tese para t_a é o mesmo do tempo mínimo de verde.

3. O tempo de verde não pode ultrapassar um valor máximo m :

$$s_{i2} - s_{j1} \leq m \quad \text{para } i = j \text{ e } i, j = 1, 2,$$

Esta restrição evita que o semáforo permaneça com a indicação semafórica em verde para arterial indefinidamente, prejudicando desta maneira as vias transversais.

4. O *offset* entre os semáforos S_i não pode ultrapassar um valor máximo p_{ij} :

$$\begin{aligned} s_{11} - s_{21} &\leq p_{12} \\ s_{21} - s_{11} &\leq p_{21}. \end{aligned}$$

É necessário garantir que o *offset* não ultrapasse um valor máximo para que todos os semáforos recebam a indicação semafórica verde na direção da arterial pelo menos uma vez a cada ciclo. A restrição é definida nas duas direções da via pois não se sabe *a priori* qual será o semáforo de referência.

O conjunto de restrições vai gerar uma especificação, representada pelo semimódulo $\mathcal{K} = \{z \in \mathbb{Z}_{\max}^n \mid z_i - z_j \leq w_{ij}, \forall i, j; 1 \leq i, j \leq n\}$ onde $W = (w_{ij})$ é uma matriz com entradas em $\mathbb{Z}_{\max} \cup \{+\infty\}$. O semimódulo também pode ser representado como $\mathcal{K} = \{z \in \mathbb{Z}_{\max}^n \mid Ez \leq z\}$ onde $E = (-W)^T$.

Após calcular a especificação é possível definir o problema de controle, que é computar o conjunto máximo de estados iniciais \mathcal{K}^\uparrow para o qual exista uma seqüência de vetores de controle que façam o estado do sistema permanecer sempre em \mathcal{K} , ou seja, não viole a especificação. Quando é possível calcular \mathcal{K}^\uparrow diz-se que \mathcal{K}^\uparrow é um semimódulo geometricamente (A,B)-invariante.

Conforme descrito no Capítulo 5, se a matriz E é irredutível o semimódulo \mathcal{K} tem volume finito. O fato de \mathcal{K} ter volume finito garante que a busca por um semimódulo geometricamente (A,B)-invariante sempre estabiliza em um número finito de passos. Quando a matriz E é irredutível sabe-se que o seu grafo de precedência $G(E)$ é fortemente conexo.

Sejam $z = [s_{11} \ s_{12} \ x_{11} \ x_{12} \ s_{21} \ s_{22} \ x_{21} \ x_{22}]^T$ e $\mathcal{K} = \{z \in \mathbb{Z}_{\max}^n \mid Ez \leq z\}$. As matrizes W e $E = (-W)^T$, para o sistema mínimo composto de duas interseções, são apresentadas nas Tabelas 6.1 e 6.2, respectivamente.

O grafo de precedência $G(E)$ para a matriz da Tabela 6.2 é mostrado na Figura 6.11. É possível perceber que o grafo encontrado é fortemente conexo, ou seja, a sua matriz E é irredutível. É possível

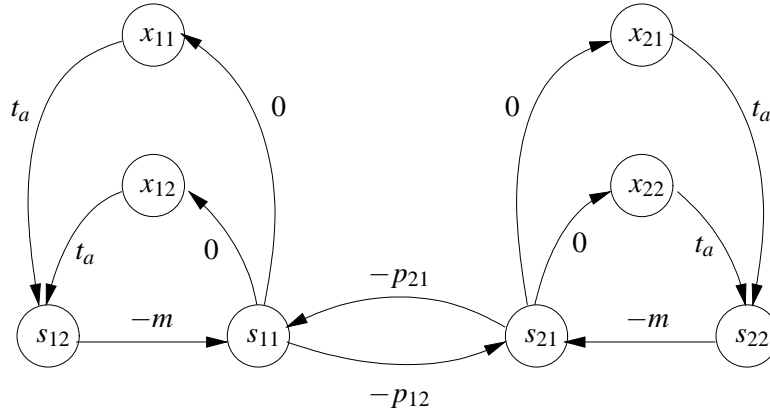
$+\infty$	$+\infty$	0	0	p_{12}	$+\infty$	$+\infty$	$+\infty$
m	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$
$+\infty$	$-t_a$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$
$+\infty$	$-t_a$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$
p_{21}	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	0	0
$+\infty$	$+\infty$	$+\infty$	$+\infty$	m	$+\infty$	$+\infty$	$+\infty$
$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$-t_a$	$+\infty$	$+\infty$
$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$-t_a$	$+\infty$	$+\infty$

Tabela 6.1: Matriz W para especificação

$-\infty$	$-m$	$-\infty$	$-\infty$	$-p_{21}$	$-\infty$	$-\infty$	$-\infty$
$-\infty$	$-\infty$	t_a	t_a	$-\infty$	$-\infty$	$-\infty$	$-\infty$
0	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$
0	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$
$-p_{12}$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-m$	$-\infty$	$-\infty$
$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	t_a	t_a
$-\infty$	$-\infty$	$-\infty$	$-\infty$	0	$-\infty$	$-\infty$	$-\infty$
$-\infty$	$-\infty$	$-\infty$	$-\infty$	0	$-\infty$	$-\infty$	$-\infty$

Tabela 6.2: Matriz $E = (-W)^T$

concluir que se a especificação para o sistema for definida com os quatro conjuntos de restrições definidos nesta tese sempre será possível encontrar um semimódulo \mathcal{K}^\uparrow geometricamente (A,B)-invariante, pois $G(E)$ sempre terá a mesma estrutura e será fortemente conexo. A solução encontrada é uma matriz K^\uparrow , sendo $\mathcal{K}^\uparrow = \text{Im}(K^\uparrow)$. O cálculo de \mathcal{K}^\uparrow é realizado através do método proposto por Katz [2007], que foi apresentado na seção 5.3.

Figura 6.11: Grafo de precedência para matriz E da especificação

Após encontrar um semimódulo geometricamente (A,B)-invariante deseja-se implementar o controle através de uma realimentação de estados $u = F z(k-1)$ onde $F \in \mathbb{Z}_{\max}^{q \times n}$ existe se \mathcal{K}^\uparrow é algebricamente (A,B)-invariante em relação ao sistema em malha fechada $z(k) = (A \oplus BF)z(k-1)$. Em outras palavras, o controle sobre os sinais semafóricos será uma função dos instantes de ocorrência dos eventos indicados pelas variáveis de estado z_i . Na álgebra convencional sempre que é possível calcular um espaço (A,B)-invariante é possível encontrar uma matriz F que realize a realimentação de estados, entretanto, quando se trabalha com a álgebra max-plus nem sempre é possível encontrar

uma matriz de realimentação F .

O grafo de eventos da Figura 6.12 representa o sistema controlado com realimentação de estados da Figura 6.10. Neste caso a matriz F tem dimensão 4×8 , ou seja, cada linha i da matriz representa uma ligação entre as transições do sistema com uma entrada de controle $u_i \in (u_1 u_2 u_3 u_4)$. Em Katz [2007] são apresentados métodos para calcular a matriz F , sendo que alguns deles estão implementados no *Max Plus Toolbox* do *Scilab* [Max Plus, 1998].

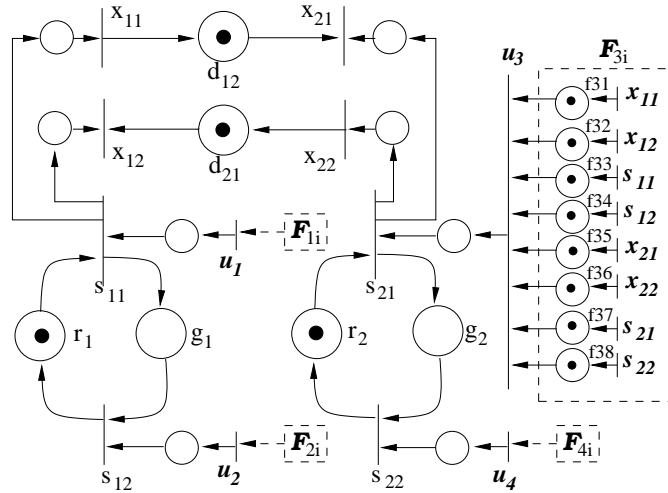


Figura 6.12: Malha viária controlada

A proposta de modelagem e controle foi apresentada para o sistema mínimo, com duas interseções, mas a extensão para um sistema com mais interseções é direto, como será mostrado na próxima seção.

6.2.1 Modelo Genérico para n Interseções

A seção anterior apresentou a proposta de modelagem e controle para o sistema mínimo, composto de 2 interseções. Nesta seção o modelo é estendido para a arterial da Figura 6.13, que é composta de n interseções semaforizadas.

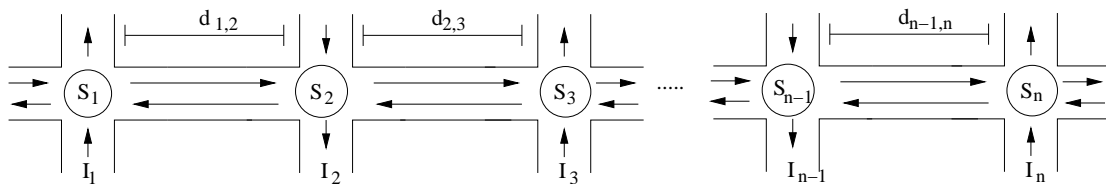
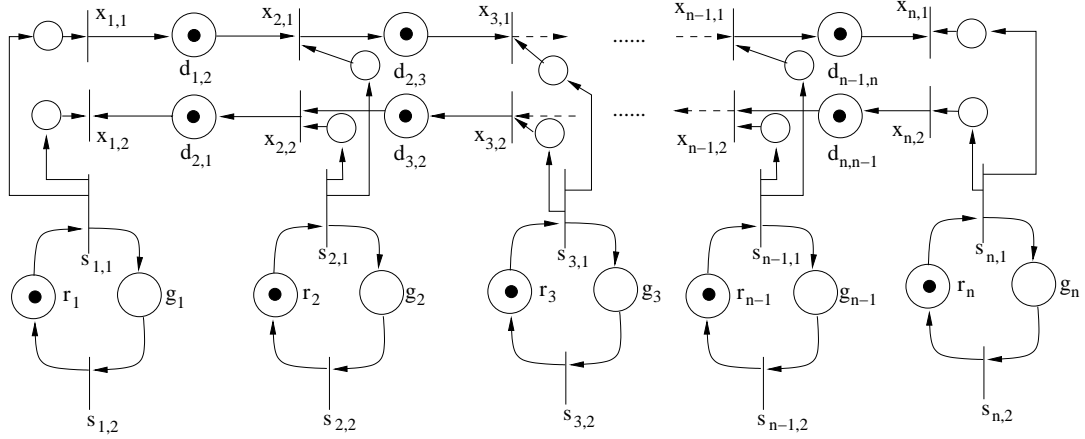


Figura 6.13: Malha viária com n interseções

O grafo de eventos semaforizado que representa a arterial da Figura 6.13 é apresentado na Figura 6.14. A arterial é modelada através de uma rede de Petri com $2n$ transições, sendo n transições para cada direção da via, e são definidos $2(n-1)$ lugares, sendo $n-1$ para cada direção. Cada semáforo é modelado conforme o modelo apresentado na Figura 6.7, sendo que cada semáforo é conectado com uma direção da via através da conexão definida na Figura 6.8, resultando em n modelos para os semáforos e $2n$ conexões (n para cada direção da via).

Figura 6.14: grafo de eventos para arterial com n interseções

A matriz A que representa o sistema max-plus linear possui dimensão $4n \times 4n$, sendo n o número de interseções. Este número corresponde a soma das $2n$ transições que representam as duas direções da via com as $2n$ transições que representam as trocas semafóricas.

No modelo apresentado não são definidas as entradas de controle pois a definição de quais semáforos controlar depende da aplicação desejada. Sendo assim, a dimensão da matriz B é variável pois o número de colunas depende do número de entradas de controle, sabe-se apenas que o número de linhas é $4n$.

Os quatro conjuntos de restrições definidos na seção anterior também podem ser generalizados para um sistema com n interseções, conforme apresentado a seguir:

1. Um pelotão atravessa uma interseção apenas quando o semáforo está verde:

$$\begin{aligned} s_{i1}(k) - x_{i1}(k) &\leq 0 \\ s_{i1}(k) - x_{i2}(k) &\leq 0 \quad i = 1, \dots, n. \end{aligned}$$

2. O tempo de verde deve ser capaz de absorver um pelotão (t_{ai} é o tempo de absorção):

$$\begin{aligned} x_{i1}(k) - s_{i2}(k) &\leq -t_{ai} \\ x_{i2}(k) - s_{i2}(k) &\leq -t_{ai} \quad i = 1, \dots, n. \end{aligned}$$

3. O tempo de verde não pode ultrapassar um máximo m_i :

$$s_{i2}(k) - s_{i1}(k) \leq m_i \quad i = 1, \dots, n.$$

4. O *offset* entre os semáforos S_i não pode ultrapassar um valor máximo p :

$$\begin{aligned} s_{i,1}(k) - s_{i+1,1}(k) &\leq p_{i,i+1} \\ s_{i+1,1}(k) - s_{i,1}(k) &\leq p_{i+1,i} \quad i = 1, \dots, (n-1). \end{aligned}$$

Após definir as restrições é possível obter a matriz E que representa a especificação, e sabe-se que a dimensão da matriz E é igual da matriz A , no caso $4n \times 4n$. A dimensão da matriz K^\dagger é variável, sabe-se apenas que ela contém $4n$ linhas. O mesmo ocorre com a matriz de controle F , onde o número de linhas depende do número de entradas de controle, sabe-se apenas que o número de colunas é $4n$.

A próxima seção apresenta a obtenção do modelo max-plus para uma arterial composta de 3 interseções, bem como a obtenção do controle para o sistema modelado.

6.2.2 Exemplo

O modelo genérico apresentado na seção 6.2.1 é aplicado na malha viária da Figura 6.15, onde $n = 3$, ou seja, a arterial possui três interseções.

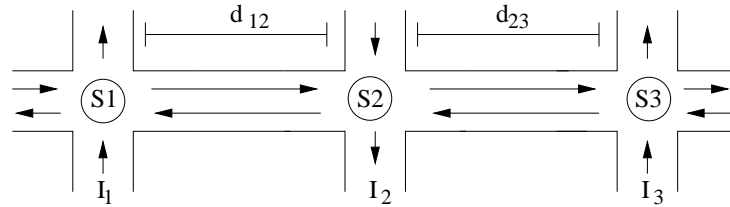


Figura 6.15: Malha viária com 3 interseções

O grafo de eventos que modela o sistema com entradas de controle está na Figura 6.16. Para resolver este exemplo, o controle foi aplicado apenas nas transições que indicam a abertura do sinal vermelho, ou seja, controla-se apenas a duração do sinal verde. Esta simplificação é realizada para diminuir a complexidade computacional do cálculo de \mathcal{K}^\dagger , e não é uma limitação teórica da proposta apresentada anteriormente.

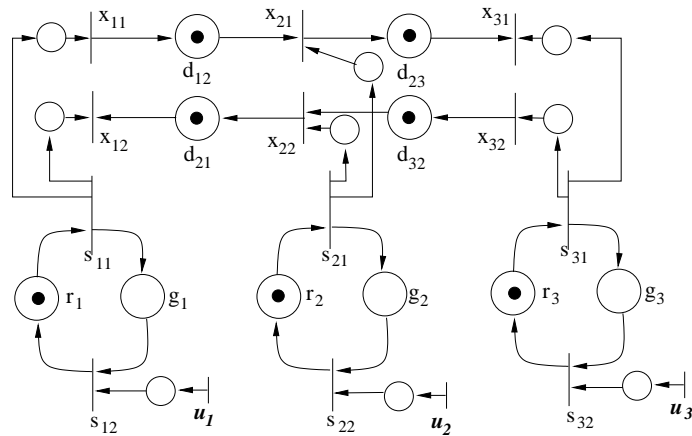


Figura 6.16: Modelo com entradas de controle

Sendo $z(k) = [x_{11}(k) \ x_{12}(k) \ s_{11}(k) \ s_{12}(k) \ x_{21}(k) \ x_{22}(k) \ s_{21}(k) \ s_{22}(k) \ x_{31}(k) \ x_{32}(k) \ s_{31}(k) \ s_{32}(k)]^T$, $u(k) = [u_1(k) \ u_2(k) \ u_3(k)]^T$, $\varepsilon = -\infty$ e $e = 0$, a matriz A do sistema $z(k) = Az(k-1) \oplus Bu(k)$ da Figura 6.16 é apresentada na Tabela 6.3 e a matriz B na Tabela 6.4.

A especificação para a malha composta de três interseções é composta pelos quatro conjuntos de restrições definidos anteriormente:

2. O tempo de verde deve ser capaz de absorver um pelotão (t_{ai} é o tempo de absorção):

$$x_{11}(k) - s_{12}(k) \leq -t_{a1}$$

$$x_{12}(k) - s_{12}(k) \leq -t_{a1}$$

$$x_{21}(k) - s_{22}(k) \leq -t_{a2}$$

$$x_{22}(k) - s_{22}(k) \leq -t_{a2}$$

$$x_{31}(k) - s_{32}(k) \leq -t_{a3}$$

$$x_{32}(k) - s_{32}(k) \leq -t_{a3}$$

3. O tempo de verde não pode ultrapassar um máximo m :

$$s_{12}(k) - s_{11}(k) \leq m_1$$

$$s_{22}(k) - s_{21}(k) \leq m_2$$

$$s_{32}(k) - s_{31}(k) \leq m_3$$

4. O *offset* entre os semáforos S_i não pode ultrapassar um valor máximo p_{ij} :

$$s_{11}(k) - s_{21}(k) \leq p_{12}$$

$$s_{21}(k) - s_{11}(k) \leq p_{12}$$

$$s_{21}(k) - s_{31}(k) \leq p_{23}$$

$$s_{31}(k) - s_{21}(k) \leq p_{23}$$

As restrições representam um semimódulo $\mathcal{K} = \{z \in \mathbb{Z}_{\max}^n | z_i - z_j \leq w_{ij}, \forall 1 \leq i, j \leq n\}$, onde $W = (w_{ij})$ é uma matriz com componentes em $\mathbb{Z}_{\max} \cup \{+\infty\}$. A matriz W para este exemplo é apresentada na Tabela 6.5.

$+\infty$	$+\infty$	$+\infty$	$-t_{a1}$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$
$+\infty$	$+\infty$	$+\infty$	$-t_{a1}$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$
0	0	$+\infty$	$+\infty$	$+\infty$	$+\infty$	p_{12}	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$
$+\infty$	$+\infty$	m_1	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$
$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$-t_{a2}$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$
$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$-t_{a2}$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$
$+\infty$	$+\infty$	p_{12}	$+\infty$	0	0	$+\infty$	$+\infty$	$+\infty$	$+\infty$	p_{23}	$+\infty$	$+\infty$	$+\infty$
$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	m_2	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$
$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$-t_{a3}$	$+\infty$
$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$-t_{a3}$	$+\infty$
$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	p_{23}	$+\infty$	0	0	$+\infty$	$+\infty$	$+\infty$	$+\infty$
$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	m_3	$+\infty$	$+\infty$	$+\infty$

Tabela 6.5: Matriz W para exemplo de três interseções

As linhas da matriz representam as restrições apresentadas anteriormente, por exemplo, a primeira linha da matriz representa $x_{11}(k) - s_{12}(k) \leq -t_{a1}$, a segunda $x_{12}(k) - s_{12}(k) \leq -t_{a1}$ e assim por diante. Como já foi citado anteriormente, podemos expressar $\mathcal{K} = \{z \in \mathbb{Z}_{\max}^n | Ez \leq z\}$ onde $E = (-W)^T$. A matriz E para este exemplo está expressa na Tabela 6.6.

Sendo $\mathcal{K} = \text{Im}E^* = I \oplus E \oplus \dots \oplus E^{n-1}$ (neste caso $*$ representa o operador estrela de Kleene) e $\mathcal{B} = \text{Im}B$, a seguinte seqüência de semimódulos é calculada para encontrar \mathcal{K}^\dagger :

$$\begin{aligned}\mathcal{X}_1 &= \mathcal{K} \\ \mathcal{X}_2 &= \mathcal{X}_1 \cap A^{-1}(\mathcal{X}_1 \ominus \mathcal{B}).\end{aligned}$$

Os algoritmos utilizados para encontrar \mathcal{K}^\dagger são implementados como filtros que são executados no *Scilab* [Max Plus, 1998].

Como se verifica que semimódulo $\mathcal{X}_2 \subsetneq \mathcal{X}_1$ é necessário calcular \mathcal{X}_3 :

$$\mathcal{X}_3 = \mathcal{X}_2 \cap A^{-1}(\mathcal{X}_2 \ominus \mathcal{B}).$$

Verifica-se que $\mathcal{X}_3 \subseteq \mathcal{X}_2$, portanto $\mathcal{K}^\dagger = \mathcal{X}_3$. A matriz da Tabela 6.7 é uma base ou geradora de \mathcal{K}^\dagger , onde $\mathcal{K}^\dagger = \text{Im}K^\dagger$, e cada coluna da matriz representa um possível estado inicial $z(0)$ para o sistema.

31	170	181	186	160	181	186	0	5	181	5	186
96	214	246	256	204	246	256	44	70	246	70	256
31	170	181	212	160	181	186	0	5	181	5	186
31	170	181	186	160	181	212	0	5	181	5	186
36	165	186	186	165	186	186	5	10	186	10	186
101	209	251	251	209	251	251	49	75	251	75	251
36	165	186	186	165	186	186	5	10	207	10	186
36	165	207	186	165	186	186	5	10	186	10	186
71	125	146	146	125	146	146	45	45	146	45	146
141	169	211	211	169	211	211	89	115	211	115	211
71	125	146	146	125	146	146	45	45	146	71	146
71	125	146	146	125	146	146	45	71	146	45	146

Tabela 6.7: Matriz com conjunto de estados iniciais K^\dagger

É possível, ainda, verificar que o semimódulo \mathcal{K}^\dagger encontrado é algebricamente (A,B)-invariante, e portanto é possível gerar ao menos uma matriz F de realimentação de estados. Para o exemplo desenvolvido foram encontradas duas matrizes F que podem atuar como controladores para o sistema. As matrizes F_1 e F_2 encontradas estão nas Tabelas 6.8 e 6.9. Os métodos para calcular F estão implementados no *Scilab* [Max Plus, 1998].

129	85	129	129	129	85	129	129	89	45	89	89
124	80	124	124	129	85	129	129	89	45	85	85
89	45	89	89	94	50	94	94	129	85	129	129

Tabela 6.8: Matriz de controle F_1

Sendo $z(0)$ a primeira coluna da matriz K^\dagger e F_1 a matriz de controle, obtém-se a seguinte evolução

$$\begin{vmatrix} 108 & 64 & 108 & 108 & 103 & 59 & 103 & 103 & 63 & 19 & 63 & 63 \\ 103 & 59 & 103 & 103 & 108 & 64 & 108 & 108 & 68 & 24 & 68 & 68 \\ 63 & 19 & 63 & 63 & 68 & 24 & 68 & 68 & 108 & 64 & 108 & 108 \end{vmatrix}$$
Tabela 6.9: Matriz de controle F_2

para o sistema $z(k)$:

$$\begin{aligned} z(0) &= [31 \ 96 \ 31 \ 31 \ 36 \ 101 \ 36 \ 36 \ 71 \ 141 \ 71 \ 71]^T \\ z(1) &= [116 \ 186 \ 116 \ 116 \ 121 \ 186 \ 121 \ 121 \ 161 \ 226 \ 161 \ 161]^T \\ z(2) &= [206 \ 271 \ 206 \ 206 \ 206 \ 271 \ 206 \ 206 \ 246 \ 311 \ 246 \ 246]^T \\ z(3) &= [291 \ 356 \ 291 \ 291 \ 291 \ 356 \ 291 \ 291 \ 331 \ 396 \ 331 \ 331]^T \\ &\vdots \\ z(k) &= (A \oplus BF)z(k-1) \end{aligned}$$

O sistema entra em regime permanente a partir da terceira iteração ($z(2)$) e o tempo de ciclo da malha viária é de 85 segundos, onde o tempo de verde estabiliza em 65 segundos em todos os semáforos.

6.3 Implementação do Controle

Para implementar a matriz F na forma de um controle em malha fechada em aplicações reais duas questões devem ser analisadas:

- a primeira questão é como calcular os vetores $x(k)$ do sistema $x(k) = Ax(k-1) \oplus Bu(k)$, sendo que os eventos acontecem à medida que o tempo evolui;
- a segunda questão é como garantir que as restrições serão respeitadas caso ocorra alguma perturbação no sistema e os tempos de viagem dos pelotões sejam alterados.

6.3.1 Obtenção dos Vetores $x(k)$

Quando se deseja calcular a evolução $x(k) = Ax(k-1) \oplus Bu(k)$ de um sistema, à medida em que os eventos acontecem, é necessário definir uma estratégia para ler as transições dos estados $x(k-1)$ do sistema. As transições do vetor de estados x podem ser divididas em dois grupos:

1. Transições que modelam as trocas semaforicas.
2. Transições que indicam a passagem de um pelotão através de uma interseção.

Eventos do primeiro grupo são mais facilmente medidos que os do segundo grupo, e isto ocorre porque os chaveamentos dos sinais semafóricos são eventos físicos facilmente identificáveis, enquanto os eventos que definem a chegada de um novo pelotão não é tão evidente. É necessário definir uma estratégia para indicar quando um grupo de veículos é considerado um novo pelotão que chega na interseção. Quando se definem os modelos max-plus para os sistemas de tráfego os pelotões são considerados um ponto, entretanto quando se trabalha com modelos reais isto não é possível. Uma estratégia será definida na seção 6.4, quando resultados experimentais serão apresentados.

Além de ler os instantes de disparo das transições $x_j \in x$, é necessário definir como calcular as entradas de controle $u_i(k)$ considerando que cada transição ocorre independentemente a medida que o sistema evolui. Cada entrada de controle $u_i(k) \in u$ é relacionada com uma linha F_i da matriz de controle F . Sejam F uma matriz $r \times n$, $x = [x_1, x_2, \dots, x_n]^T$ e $u = [u_1, u_2, \dots, u_n]^T$ vetores, cada entrada de controle $u_i \in u$ pode ser calculada da seguinte maneira:

$$\begin{aligned} u_i(k) &= F_i x(k-1), \\ &= F_{i1}x_1(k-1) \oplus \dots \oplus F_{in}x_n(k-1), \\ &= \bigoplus_{j=1}^n F_{ij} \otimes x_j(k-1). \end{aligned}$$

A obtenção de u_i pode ser realizada usando uma estrutura de dados do tipo pilha. O cálculo de $u_i(k)$ é atualizado quando as transições $x_j(k-1)$ são disparadas. Quando uma transição $x_j(k-1) \in x$ é sensibilizada, a operação $F_{ij} \otimes x_j(k-1)$ é executada e o topo da pilha recebe \oplus (max) entre o seu valor corrente e o novo valor calculado. Portanto, após todas as n transições de $x(k-1)$ terem sido lidas, o topo da pilha contém $u_i(k)$. O algoritmo apresentado a seguir mostra um procedimento para calcular $u_i(k)$, onde os parâmetros de entrada são a matriz de controle F e o índice i da entrada de controle que se deseja calcular.

procedure CONTROL-INPUT(F, i)

$i \leftarrow i;$

▷ índice da entrada de controle i

$u_i \leftarrow 0;$

$aux \leftarrow 0;$

$stack \leftarrow \emptyset;$

▷ pilha vazia

for $l = 1 : n$ **do**

▷ copia linha i de F

$F_i(1, l) \leftarrow F(i, l);$

for $l = 1 : n$ **do**

▷ lê x_j e atualiza a pilha

Read $x_j(k-1) \in x;$

$aux \leftarrow F_i(1, j) \otimes x_j(k-1);$

$stack \leftarrow aux \oplus stack;$

$u_i \leftarrow stack;$

return $u_i;$

end procedure

A causalidade é garantida pela estrutura do grafo de eventos temporizado com realimentação de estados. O k -ésimo disparo de u_i somente acontece após o disparo de todas as transições $x_j(k-1)$, com $x_j \in x$.

A outra questão importante que deve ser considerada quando se deseja implementar a matriz F como um controlador em malha fechada é a robustez deste controlador. O critério de robustez que interessa nesta tese está relacionado com a variação dos tempos de viagem d_{ij} em relação aos valores nominais para d_{ij} que garantem a (A,B)-invariância do sistema.

6.3.2 Robustez do Controlador

O resultado da proposta de modelagem e controle apresentada na seção 6.2 é um conjunto de estados iniciais dado pelo semimódulo \mathcal{K}^\uparrow que garante uma especificação para o sistema, e matrizes de controle F que garantem a (A,B)-invariância para o sistema controlado com realimentação de estados. O cálculo de \mathcal{K}^\uparrow e F é realizado com valores nominais para os tempos semafóricos e tempos de viagem d_{ij} . Entretanto, sabe-se que quando se aplica a matriz F em um sistema real é possível encontrar valores para d_{ij} que diferem dos valores nominais. Para garantir que o controlador F é robusto é necessário encontrar limites para d_{ij} de modo que F continue garantindo a (A,B)-invariância para o sistema, ou seja, que o sistema evolua obedecendo a especificação desejada.

Neste trabalho a determinação dos limites para d_{ij} é obtida pelo teste exaustivo da condição $(A \oplus BF)\mathcal{K}^\uparrow \subset \mathcal{K}^\uparrow$. Esta condição garante que o conjunto de estados iniciais \mathcal{K}^\uparrow calculado para o sistema definido pela equação (5.14) é algebricamente (A,B)-invariante. Em outras palavras, a condição testa se a aplicação do controlador F em um sistema realimentado, com estado inicial $x(0) \in \mathcal{K}^\uparrow$ vai sempre evoluir para um estado onde as restrições definidas para \mathcal{K}^\uparrow são satisfeitas, ou seja, a especificação nunca falha.

Sejam A uma matriz $n \times n$, B uma matriz $n \times r$, F uma matriz de controle $r \times n$, S uma matriz $n \times p$ com um conjunto de estados iniciais e M uma matriz $n \times n$ onde $m_{ij} = 1$ se existe uma ligação entre as interseções i e j ($m_{ij} = \varepsilon$ caso contrário). O algoritmo apresentado a seguir mostra um procedimento para calcular limites para d_{ij} que garantem a (A,B)-invariância para o sistema, e seus parâmetros de entrada são as matrizes A, B, F, S e M .

procedure LIMITS (A, B, F, S, M)

$Aux \leftarrow A$;

$i \leftarrow 1$;

$t \leftarrow true$;

for $j = 1 : n$ **do**

▷ busca por entradas $m_{ij} = 1$ em M

for $k = 1 : n$ **do**

if $M[j, k] = 1$ **then**

$w(i) \leftarrow A[j, k]$;

▷ coloca $a_{ij} = d_{ij}$ em w

$y(i) \leftarrow j$;

▷ salva os índices de A

$z(i) \leftarrow k$;

$i \leftarrow i \otimes 1$;

```

while t do
  for  $l = 1 : i$  do
     $w(l) \leftarrow w(l) \otimes 1;$  ▷ incrementa  $w$  por 1
     $Aux[y(l), z(l)] \leftarrow w(l);$  ▷ atualiza  $Aux$ 
     $t \leftarrow \text{includespan}(S, (Aux \oplus BF)S)$  ▷ realiza o teste
  for  $l = 1 : i$  do
    return  $w(l) \otimes (-1)$ 
end procedure

```

No procedimento apresentado as entradas $a_{ij} = d_{ij}$ da matriz A são incrementadas em uma unidade até o momento em que o teste falha, e neste momento encontra-se o limite d_{ij}^{max} para d_{ij} que garante a (A,B)-invariância. Uma matriz auxiliar é utilizada para realizar a operação, a matriz Aux que é inicializada com A , e suas entradas correspondem aos tempos de viagem que são incrementados. O procedimento que implementa o teste $(A \oplus BF)\mathcal{K}^\dagger \subset \mathcal{K}^\dagger$ é chamado *includespan* e faz parte do *Maxplus Toolbox* para o *Scilab*.

Além disso, resultados de simulação mostram que o teste também não falha para os tempos de viagem dentro do intervalo $0 \leq d_{ij} \leq d_{ij}^{max}$. Ou seja, os valores de d_{ij} dentro do intervalo também garantem que as restrições sejam satisfeitas.

Para os casos onde o tempo de viagem é maior que o valor d_{ij}^{max} calculado, é possível encontrar uma de duas situações:

1. Encontra-se um conjunto de estados iniciais $\mathcal{K}' \subset \mathcal{K}^\dagger$ algebricamente (A,B)-invariante, e uma matriz de controle F' correspondente.
2. Encontra-se um conjunto de estados iniciais $\mathcal{K}' \subseteq \mathcal{K}^\dagger$ que é geometricamente (A,B)-invariante mas não algebricamente (A,B)-invariante, ou seja, não é possível calcular uma matriz de controle F .

Aplicando o algoritmo proposto no exemplo da seção 6.2.2 é possível definir os valores limites para d_{ij} que garantam a (A,B)-invariância. Os valores calculados são:

- $0 \leq d_{12} = d_{21} \leq 40s$ e $0 \leq d_{23} = d_{32} \leq 45s$ quando o controlador F_1 é aplicado;
- $0 \leq d_{12} = d_{21} \leq 19s$ e $0 \leq d_{23} = d_{32} \leq 24s$ quando o controlador F_2 é aplicado.

A seção 6.4 apresentará os resultados obtidos através de diversos experimentos realizados. Os algoritmos aqui apresentados são implementados para permitir a utilização dos modelos max-plus obtidos em uma simulação de um sistema de tráfego real.

6.4 Experimentos

Nesta seção são apresentados os resultados encontrados quando se aplica uma matriz de controle F e os estados iniciais \mathcal{K}^\dagger em um modelo de simulação de tráfego. O propósito é verificar se é possível transmitir os resultados dos modelos max-plus encontrados para um modelo de simulação, e observar o comportamento do sistema controlado com realimentação de estados. Os aspectos relacionados com a robustez da matriz F também são investigados.

6.4.1 Modelo de Tráfego para Simulação

É necessário definir um modelo de tráfego que permita simular as condições de tráfego e aplicar a matriz de controle F . O modelo de tráfego utilizado nesta tese é baseado no modelo proposto por Cervantes [2005]. O modelo de Cervantes [2005] é uma simplificação do modelo de tráfego do PRODYN [Farges et al., 1983].

São considerados instantes de tempo t espaçados de um período $t_s = t_{j+1} - t_j$ constante, ou seja, t é um tempo de amostra. A evolução da indicação semaforica pode ser descrita através da seguinte equação:

$$v_i(t+1) = u_i(t), \quad i = 1, 2, \dots, I \quad (6.1)$$

onde $v_i(t) \in \{0, 1\}$ é o estágio vigente na interseção i durante o intervalo de tempo (t_j, t_{j+1}) , sendo que $v_i(t) = 0$ significa que a indicação semaforica está vermelha e $v_i(t) = 1$ que a indicação semaforica está verde. A variável $u_i(t) \in \{0, 1\}$ é o valor do controle a ser aplicado no período subsequente, onde 0 é a indicação do vermelho e 1 a indicação verde. A variável I indica o número de interseções do sistema.

Uma via pode ser definida como um segmento viário urbano que liga duas interseções, além de servir de entrada ou saída para uma malha viária urbana. As vias são modelados através de variáveis que caracterizam as suas filas e os veículos trafegando em velocidade livre. O modelo de fila adotado é o de fila vertical, e a sua evolução temporal em uma aproximação ou *link* (ver Apêndice A) é dada pela seguinte equação:

$$x_l(t+1) = \max \{0, x_l(t) + a_{l1}(t) - s_l m_l(t)\} \quad (6.2)$$

onde $x_l(t)$ é a fila formada na linha de parada de uma faixa l , e a variável $a_{l1}(t)$ indica o número de veículos em velocidade livre que trafegam na primeira seção de um *link*. A taxa média de descarga da fila é dada por s_l e é considerada conhecida e constante para todos os *links*. Por fim, $m_l(t) \in \{0, 1\}$ indica o controle da faixa l , sendo $m_l(t) = u_i(t)$ para as arteriais e $m_l(t) = \overline{u_i(t)}$ para as transversais ou secundárias. Para o caso das vias de mão dupla existe uma fila vertical para cada direção, onde cada direção é representada por um *link*.

Na via de mão dupla da Figura 6.17 é possível identificar 2 interseções e 6 aproximações ou *links*. Em cada *link* $l, l = 1, \dots, 6$ forma-se uma fila vertical que é dada pela equação (6.2).

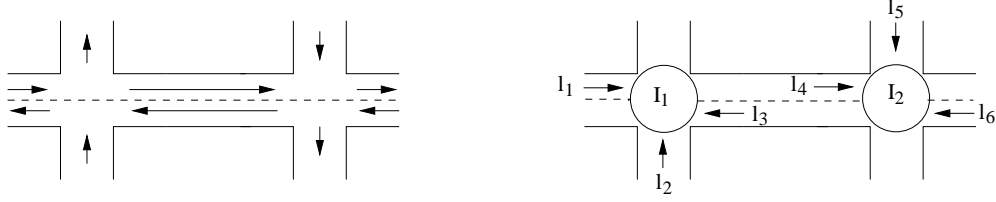


Figura 6.17: Aproximações ou *links* em via de mão dupla

O modelo de tráfego para os veículos trafegando em velocidade livre ao longo da via é um modelo discretizado. A discretização ocorre através da criação de seções em uma faixa, onde o tempo necessário para atravessar uma seção é igual a um período de amostragem t_s multiplicado pela velocidade de cruzeiro (considerada constante). Ou seja, o número de seções é definido através da seguinte equação:

$$\mathcal{N}(l) = \frac{L_l}{v_l t_s} \quad (6.3)$$

onde $\mathcal{N}(l)$ é o número de seções, L_l é o comprimento de cada faixa l , v_l é a velocidade livre de percurso e t_s é o tamanho do período de amostragem. O número de seções $\mathcal{N}(l)$ é definido de tal forma que passagem de uma seção para a seguinte é feita exatamente em um período amostral, como mostra a Figura 6.18.

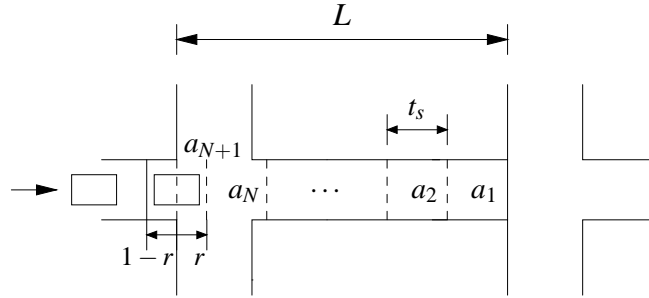


Figura 6.18: Seções de uma faixa

As equações que descrevem a ocupação $a_l(t+1)$ ao longo das seções de uma faixa são:

$$a_{l,j}(t+1) = a_{l,j+1}(t) \quad j = 1, \dots, N_l - 1 \quad (6.4)$$

$$a_{l,N_l}(t+1) = a_{l,N_l+1}(t) + (1 - r_l)z_l(t) \quad (6.5)$$

$$a_{l,N_l+1}(t+1) = r_l z_l(t) \quad (6.6)$$

onde $z_l(t)$ é o fluxo de veículos no início da faixa a montante da interseção. N_l e r_l são, respectivamente, a parte inteira e fracionário de \mathcal{N}_l .

O fluxo de veículos $z_l(t)$ é modelado de duas formas. Para as faixas alimentadas por filas a

montante tem-se as seguintes equações:

$$z_l(t) = \sum_{l' \in U_l} p_{l'l} \times \min(x_{l'}(t) + a_{l',1}(t), s_{l'} m_{l'}(t)) \quad (6.7)$$

onde $l' \in U_l$ são as faixas pertencentes a U_l , que é o conjunto de faixas que alimentam uma determinada faixa l . A variável $p_{l'l}$ é a proporção de veículos que convergem das filas para a faixa a jusante, considerada conhecida e constante.

Para o caso das faixas de entrada, o fluxo $z_l(t)$ não é calculado através das equações apresentadas. Nestes casos $z_l(t)$ é simplesmente o resultado da contagem veicular.

As próximas seções apresentam os resultados encontrados para a simulação do exemplo de uma malha de três interseções desenvolvido na seção 6.2.

6.4.2 Algoritmo para Simulação

As simulações realizadas buscam aplicar os resultados dos modelos max-plus das arteriais no modelo de tráfego para simulação apresentado na seção anterior, e é possível definir um algoritmo que exprima esta idéia.

Simula-se uma via arterial composta por três interseções. A Figura 6.19 apresenta um diagrama com os *links* definidos para a arterial.

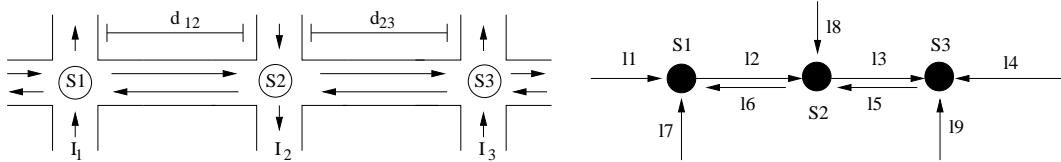


Figura 6.19: Exemplo de três interseções e diagrama de *links*

É possível definir uma relação entre os *links* da arterial com as transições da rede de Petri definida no modelo max-plus. A rede de Petri para o exemplo é apresentada na Figura 6.16. Para encontrar a relação são necessárias algumas definições:

- os *links* são representados pelo vetor $l = [l_1 \ l_2 \ l_3 \ l_4 \ l_5 \ l_6 \ l_7 \ l_8 \ l_9]$;
- os controles dos *links* são representados pelo vetor $m = [m_{11} \ m_{12} \ m_{13} \ m_{14} \ m_{15} \ m_{16} \ m_{17} \ m_{18} \ m_{19}]$, onde $m_{li} \in \{0, 1\}, i = 1 \dots 9$. O valor 0 significa que a indicação semafórica está vermelha e 1 significa que a indicação está verde para o *link* l_i ;
- as transições da rede de Petri são representadas pelo vetor de estados $k = [s_{11} \ s_{12} \ x_{11} \ x_{12} \ s_{21} \ s_{22} \ x_{21} \ x_{22} \ s_{31} \ s_{32} \ x_{31} \ x_{32}]$.

A transição x_{11} da rede de Petri é sensibilizada quando o semáforo S_1 recebe a indicação semafórica verde, ou seja, quando a transição s_{11} é sensibilizada. Analisando os vetores apresentados pode-se relacionar o chaveamento de 0 para 1 em $m_{l1} \in m$ com os disparos das transições s_{11} e x_{11} na rede de Petri, e este procedimento também pode ser feito com as outras transições. Ou seja, é possível identificar no modelo de tráfego as transições da rede de Petri, permitindo assim a representação dos modelos max-plus encontrados através do modelo de tráfego definido na seção anterior. As seguintes relações podem ser estabelecidas:

- o *link* l_1 está relacionado com a transição x_{11} e s_{11} ;
- o *link* l_2 está relacionado com a transição x_{21} e s_{21} ;
- o *link* l_3 está relacionado com a transição x_{31} e s_{31} ;
- o *link* l_4 está relacionado com a transição x_{12} e s_{31} ;
- o *link* l_5 está relacionado com a transição x_{22} e s_{21} ;
- o *link* l_6 está relacionado com a transição x_{32} e s_{11} .

Como é possível identificar as transições da rede de Petri é possível aplicar os algoritmos definidos na seção 6.3 para calcular as entradas de controle u da rede de Petri. No caso do modelo de tráfego os controles $u_j, j = 1 \dots 3$ são os instantes de abertura do sinal vermelho dos *links*, ou seja, são os instantes em que os valores dos controles $m_{li} \in m$ são chaveados de 1 para 0, e como o tempo de vermelho é constante é possível definir a abertura do sinal verde também. Deseja-se aplicar o controle F para calcular os instantes de abertura do vermelho para arterial, e assim garantir que as restrições serão sempre respeitadas. A estrutura de pilha é utilizada e a detecção da chegada de um pelotão é definida pela inspeção da evolução dos veículos através das seções definidas para via. A identificação do pelotão se dá em dois casos:

- caso exista fila na linha de parada do *link* l_i no instante da abertura do verde considera-se que a transição x_{ij} correspondente na rede de Petri foi sensibilizada;
- caso não exista fila, espera-se a chegada do primeiro veículo que partiu da linha de parada do *link* a montante, e quando isto acontece armazena-se o instante, o qual corresponde a transição x_{ij} na rede de Petri.

O algoritmo apresentado a seguir mostra como é possível implementar os modelos max-plus encontrados para o exemplo no modelo de tráfego para simulação. Os parâmetros de entrada do procedimento são: um vetor k_0 com um estado inicial da rede de Petri, a matriz de controle F , e o vetor m com a indicação semafórica inicial dos *links*.

procedure SIMULA (k_0, F, m)

$k(1, 1 : 12) \leftarrow k_0$;

$m(1, 1 : 9) \leftarrow m$;


```

 $v(1,1) \leftarrow k(1,1);$  ▷ primeiro vermelho de  $S_1$ 
 $v(1,2) \leftarrow k(1,5);$  ▷ primeiro vermelho de  $S_2$ 
 $v(1,3) \leftarrow k(1,9);$  ▷ primeiro vermelho de  $S_3$ 
 $Tcontrole \leftarrow$  tempo de controle; ▷ tempo de simulação
 $r \leftarrow$  tempo de vermelho;
 $cont \leftarrow 0$  ▷ variável auxiliar
for  $j = 1 : Tcontrole$  do
  if  $j = v(j,1)$  then ▷ abertura do vermelho em  $S_1$ 
    atualiza  $m(j+1, 1 : 9)$ 
    atualiza  $k(j+1, 1 : 12)$ 
  if  $j = v(j,2)$  then ▷ abertura do vermelho em  $S_2$ 
    atualiza  $m(j+1, 1 : 9)$ 
    atualiza  $k(j+1, 1 : 12)$ 
  if  $j = v(j,3)$  then ▷ abertura do vermelho em  $S_3$ 
    atualiza  $m(j+1, 1 : 9)$ 
    atualiza  $k(j+1, 1 : 12)$ 
  if  $j = v(j,1) + r$  then ▷ abertura do verde em  $S_1$ 
    atualiza  $m(j+1, 1 : 9)$ 
    atualiza  $k(j+1, 1 : 12)$  ▷ detecta pelotão
  if  $j = v(j,2) + r$  then ▷ abertura do verde em  $S_2$ 
    atualiza  $m(j+1, 1 : 9)$ 
    atualiza  $k(j+1, 1 : 12)$  ▷ detecta pelotão
  if  $j = v(j,3) + r$  then ▷ abertura do verde em  $S_3$ 
    atualiza  $m(j+1, 1 : 9)$ 
    atualiza  $k(j+1, 1 : 12)$  ▷ detecta pelotão
  if  $j \neq v(j,1 : 3) \wedge j \neq v(j,1 : 3) + r$  then
     $m(j+1, 1 : 9) \leftarrow m(j, 1 : 9)$  ▷ não altera  $m$ 
     $k(j+1, 1 : 12) \leftarrow k(j, 1 : 12)$  ▷ não altera  $k$ 
  for  $i = 1 : 9$  do
    Calcular a geração dos fluxos  $z(j+1, i)$  para cada link
    for  $n = 1 : N$  do ▷  $N$  é o total de seções do link
      Calcular a evolução  $a(j+1, n)$  dos veículos através das seções
    Calcular a fila  $x(j+1, i)$  para cada link
  for  $y = 1 : 12$  do ▷ Cálculo do controle  $u(k)$ 
    if  $k(j,y) == j+1 \wedge cont < 12$  then
       $cont = cont + 1$ 
    if  $cont == 12$  then ▷ já detectou todos os disparos de uma iteração
      for  $i = 1 : 12$  do
         $u_1 = \max\{F(1, i) + k(j, i)\}$ 
         $u_2 = \max\{F(2, i) + k(j, i)\}$ 

```

```


$$u_3 = \max\{F(3, i) + k(j, i)\}$$


$$v(j+1, 1) \leftarrow u_1 \quad \triangleright \text{próximo vermelho de S1}$$


$$v(j+1, 2) \leftarrow u_2 \quad \triangleright \text{próximo vermelho de S2}$$


$$v(j+1, 3) \leftarrow u_3 \quad \triangleright \text{próximo vermelho de S3}$$


$$cont \leftarrow 0$$

    Atualiza as entradas de  $k(j+1)$  correspondentes as aberturas de verde e vermelho
return  $x, a, m, k$ 
end procedure

```

O algoritmo permite gerar gráficos que mostram a evolução a_l dos veículos ao longo dos l links, as filas x_l nos links, bem como as trocas semafóricas armazenadas na matriz m . Como todas as iterações da rede de Petri são armazenadas nos vetores k é possível plotar gráficos para verificar se as restrições foram violadas em algum momento da simulação. As seções seguintes apresentam as simulações que foram realizadas para o exemplo da seção 6.2.2, sendo que todos os programas foram implementados no *MATLAB*.

6.4.3 Simulação 1

Os dados relativos à primeira simulação são:

- estado inicial: $k_0 = [1 \ 66 \ 1 \ 1 \ 6 \ 71 \ 6 \ 6 \ 41 \ 101 \ 41 \ 41]^T$, sendo $k_0 = K^\uparrow(1 : 12) - 30$ (todos os elementos da primeira coluna de K^\uparrow descontados de 30 unidades) e $k_0 \in \mathcal{K}^\uparrow$;
- tempo de amostra de 1 segundo;
- tempo de controle de 500 segundos;
- matriz de controle:

$$F_2 = \begin{bmatrix} 129 & 85 & 129 & 129 & 129 & 85 & 129 & 129 & 89 & 45 & 89 & 89 \\ 124 & 80 & 124 & 124 & 129 & 85 & 129 & 129 & 89 & 45 & 85 & 85 \\ 89 & 45 & 89 & 89 & 94 & 50 & 94 & 94 & 129 & 85 & 129 & 129 \end{bmatrix}$$

- fluxo de entrada de veículos nos links l_1 e l_4 : $z = 0.2$ veic/s, constante no tempo;
- fluxo de saturação: $s = 0.5$ veic/s.

Os gráficos gerados pela simulação apresentam três curvas relativas aos links das arteriais: a indicação semafórica ao longo do tempo $m_i(t)$, a chegada de veículos na seção situada na linha de parada de cada link $a_{i1}(t)$ e a fila $x_i(t)$ formada na linha de parada de cada link, sendo $i = 1, \dots, 9$ o índice dos links.

O gráfico apresentado na Figura 6.20 apresenta as curvas geradas para o link l_1 . O link l_1 é uma entrada na arterial, logo o fluxo veicular $z(t)$ é constante e igual a 0.2 veic/s. É possível perceber

que nos instantes onde o semáforo está com indicação semaforica vermelha (valor 0) o fluxo veicular que chega forma uma fila vertical na linha de parada de l_1 . Quando o semáforo recebe sinal verde a fila é esgotada e torna-se nula até o próximo sinal vermelho. O conjunto de veículos que parte de l_1 representa um pelotão em direção ao *link* l_2 .

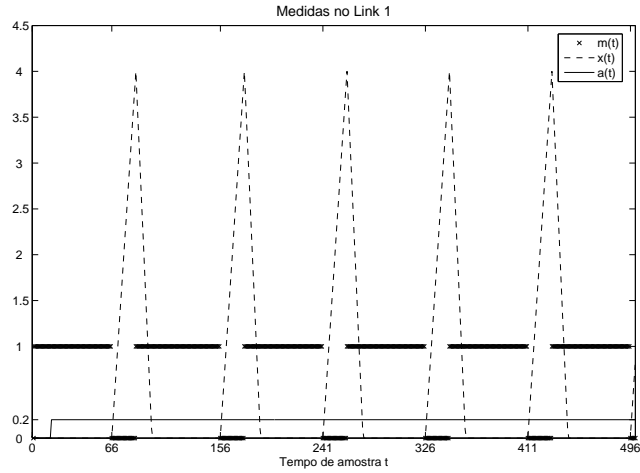


Figura 6.20: Evolução das filas $x_1(t)$ em função das chegadas $a_{11}(t)$ e da sinalização $m_1(t)$

Os gráficos das Figuras 6.21 e 6.22 apresentam as curvas geradas para os *links* l_2 e l_3 . Percebe-se que nem todos os veículos que partem de l_1 conseguem cruzar o *link* l_2 no mesmo ciclo, mas o pelotão com 0.5 veic/s passa completamente por l_2 . Existe um conjunto de veículos que encontram o semáforo S_2 com o sinal vermelho, e portanto uma fila é formada em l_2 . Como existe formação de fila em l_2 o fluxo veicular quando o semáforo recebe indicação verde é o fluxo de saturação, e este fluxo permanece até o momento que a fila é esgotada. A partir deste momento os veículos trafegam segundo o fluxo de entrada.

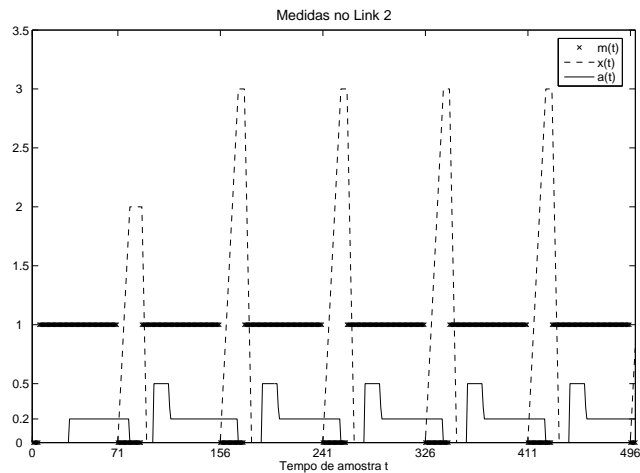


Figura 6.21: Evolução das filas $x_2(t)$ em função das chegadas $a_{21}(t)$ e da sinalização $m_2(t)$

Analisando as curvas do *link* l_3 observa-se duas chegadas distintas de grupos de veículos. É possível detectar claramente o pelotão que partiu de l_2 , e além disso um pequeno grupo de veículos.

Este grupo é formado pelos veículos que não conseguiram atravessar o *link* l_2 . Como os dois grupos de veículos chegam em l_3 durante o tempo de vermelho a fila que é formada neste *link* é maior que a formada no *link* l_2 . Neste caso o pelotão com 0.5 veic/s que partiu de l_2 chega no final do vermelho.

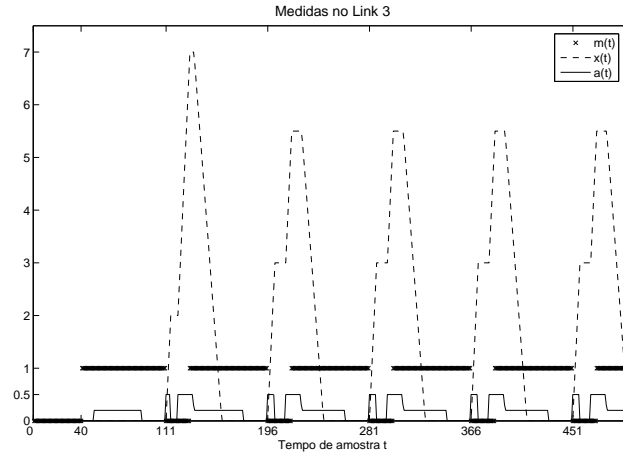


Figura 6.22: Evolução das filas $x_3(t)$ em função das chegadas $a_{31}(t)$ e da sinalização $m_3(t)$

As Figuras 6.23, 6.24 e 6.25 apresentam as curvas geradas para os *links* l_4 , l_5 e l_6 , respectivamente, onde l_4 é o *link* de entrada nesta direção da via. Percebe-se um comportamento semelhante ao explicado para os *links* l_1 , l_2 e l_3 só que agora em l_6 passa todo o pelotão. As curvas geradas para os *links* que representam as vias transversais não são apresentados pois o foco deste trabalho é o comportamento da arterial.

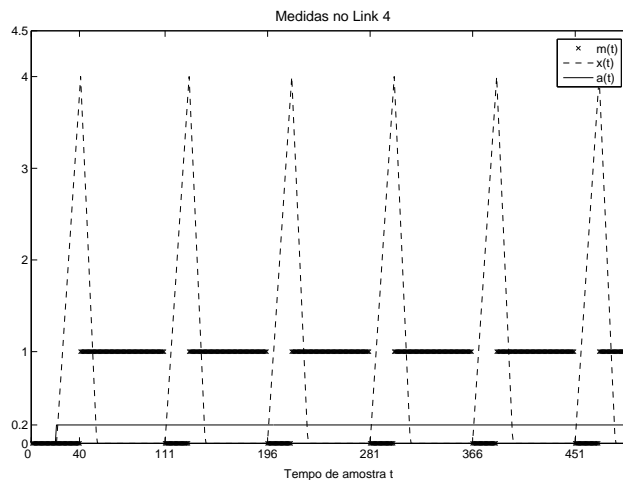


Figura 6.23: Evolução das filas $x_4(t)$ em função das chegadas $a_{41}(t)$ e da sinalização $m_4(t)$

O comportamento ideal do sistema é aquele onde todos os pelotões que partem de l_1 ou l_4 encontram todos os semáforos da arterial com a indicação verde, ou seja, não precisam parar em uma fila. Analisando os gráficos apresentados percebe-se que esta situação não acontece, entretanto é possível encontrar uma banda de verde para o exemplo desenvolvido. A Figura 6.26 apresenta as bandas geradas para ambas as direções da arterial. As linhas horizontais indicam o tempo de vermelho ao longo dos ciclos e as diagonais as bandas encontradas em ambos os sentidos da arterial.

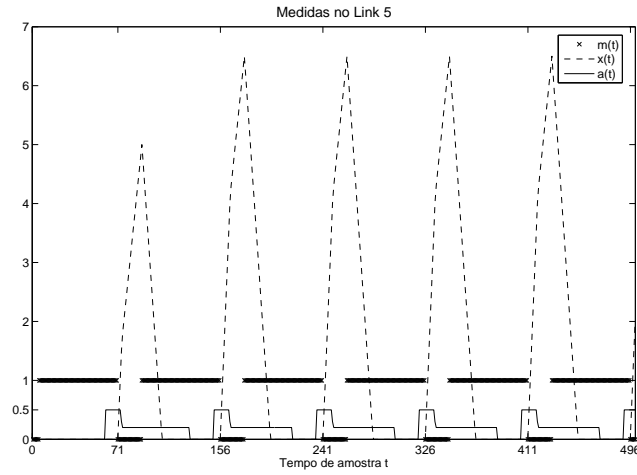


Figura 6.24: Evolução das filas $x_5(t)$ em função das chegadas $a_{51}(t)$ e da sinalização $m_5(t)$

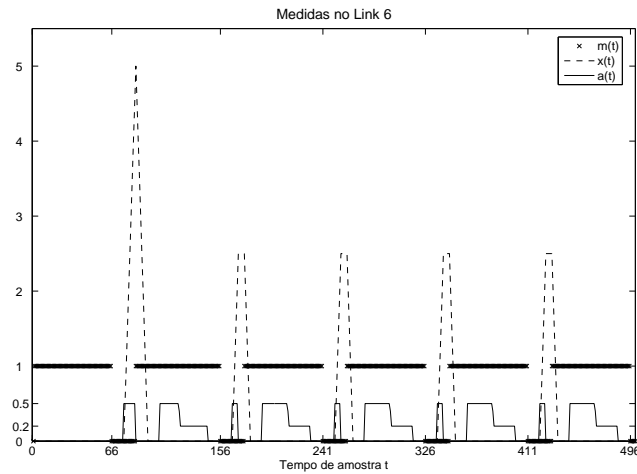


Figura 6.25: Evolução das filas $x_6(t)$ em função das chegadas $a_{61}(t)$ e da sinalização $m_6(t)$

Os veículos que partem de I_1 dentro da banda de verde vão encontrar os semáforos S_2 e S_3 com a indicação semafórica verde. O mesmo raciocínio vale para os veículos que partem de I_4 . Os veículos que trafegam fora da banda definida são os veículos que contribuem para a formação das filas que foram apresentadas nas curvas dos *links*, ou seja, são aqueles que encontram um semáforo com a indicação vermelha ao longo da sua trajetória. O comprimento das bandas encontradas foram: no sentido $I_1 \rightarrow I_3$ uma banda de 45 segundos e no sentido $I_3 \rightarrow I_1$ uma banda de 40 segundos. Aplicando o algoritmo de banda máxima desenvolvido por Carlson [2006] (uma aplicação do método de Morgan e Little [1964]) o valor da banda encontrada, utilizando os mesmos valores de tempo de ciclo e tempo de vermelho, foi 43 segundos em ambos os sentidos da arterial. O objetivo do método proposto nesta tese não é obter a banda máxima, entretanto, neste caso os resultados encontrados ficaram próximos do resultado ótimo.

O algoritmo de simulação desenvolvido permite gerar gráficos que mostram se as restrições definidas para gerar o conjunto de estados iniciais \mathcal{K}^\dagger são sempre respeitadas. Como já foi citado

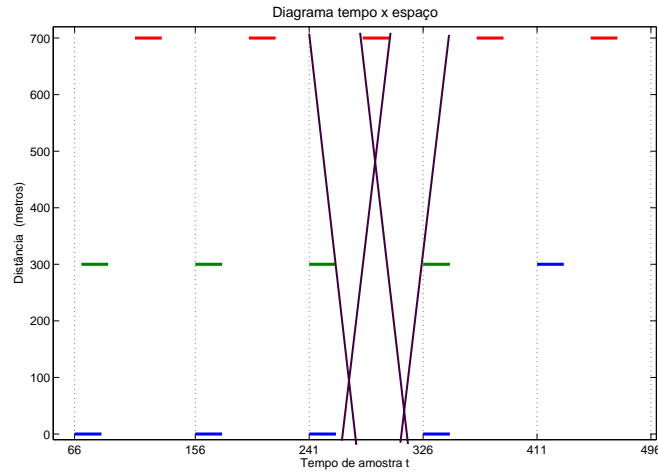


Figura 6.26: Banda de verde gerada – Simulação 1

anteriormente as restrições podem ser divididas em quatro grupos:

- grupo 1: restrições que garantem que um pelotão só cruza uma interseção com o sinal verde;
- grupo 2: restrições que garantem que os tempos de verde são suficientes para absorver um pelotão;
- grupo 3: restrições que definem um valor máximo para o tempo de verde;
- grupo 4: restrições que definem um valor máximo para o *offset*.

A Tabela 6.10 apresenta os valores encontrados para cada restrição durante a simulação. As inequações apresentadas na primeira coluna da tabela representam os quatro grupos de restrições que definem a especificação \mathcal{K} . As outras colunas apresentam os resultados encontrados para k iterações, onde $k = 1, \dots, 5$.

A análise dos valores encontrados revela que os limites definidos pelas restrições nunca foram ultrapassados, o que garante que as restrições são sempre respeitadas. Entretanto, uma questão que pode ser levantada quando se analisa o conjunto de restrições do grupo 2 é o fato dos pelotões formados nos *links* l_1 e l_4 não conseguirem atravessar a via sem serem interceptados, ou seja, sempre existem alguns veículos que encontram o semáforo vermelho, e isto poderia ser interpretado como uma violação das restrições. Na verdade a restrição é respeitada pois ela foi definida com o tempo de absorção igual ao tempo mínimo de verde (44 segundos).

Os resultados encontrados nesta simulação foram satisfatórios pois mostram que é possível simular em um modelo de tráfego os modelos max-plus obtidos para o exemplo explorado. Utiliza-se como estado inicial da rede um estado encontrado no modelo max-plus, e verifica-se que a especificação definida é sempre garantida.

Restrições	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$
$s_{11}(k) - x_{11}(k) \leq 0$	0	0	0	0	0
$s_{11}(k) - x_{12}(k) \leq 0$	-20	-15	-15	-15	-15
$s_{21}(k) - x_{21}(k) \leq 0$	-10	-15	-15	-15	-15
$s_{21}(k) - x_{22}(k) \leq 0$	0	0	0	0	0
$s_{31}(k) - x_{31}(k) \leq 0$	0	0	0	0	0
$s_{31}(k) - x_{32}(k) \leq 0$	0	0	0	0	0
$x_{11}(k) - s_{12}(k) \leq -44$	-70	-65	-65	-65	-65
$x_{12}(k) - s_{12}(k) \leq -44$	-50	-50	-50	-50	-50
$x_{21}(k) - s_{22}(k) \leq -44$	-55	-50	-50	-50	-50
$x_{22}(k) - s_{22}(k) \leq -44$	-65	-65	-65	-65	-65
$x_{31}(k) - s_{32}(k) \leq -44$	-65	-65	-65	-65	-65
$x_{32}(k) - s_{32}(k) \leq -44$	-65	-65	-65	-65	-65
$s_{11}(k) - s_{21}(k) \leq 5$	-5	0	0	0	0
$s_{21}(k) - s_{11}(k) \leq 5$	5	0	0	0	0
$s_{21}(k) - s_{31}(k) \leq 40$	-40	-40	-40	-40	-40
$x_{31}(k) - s_{21}(k) \leq 40$	40	40	40	40	40
$s_{12}(k) - s_{11}(k) \leq 70$	70	65	65	65	65
$s_{22}(k) - s_{21}(k) \leq 65$	65	65	65	65	65
$x_{32}(k) - s_{31}(k) \leq 70$	65	65	65	65	65

Tabela 6.10: Resultados encontrados para as restrições – Simulação 1

A próxima seção apresentará uma variação da simulação aqui desenvolvida. O estado inicial do sistema será um estado que pertence a \mathcal{K} (a especificação definida para o sistema) mas não pertence a \mathcal{K}^\uparrow , que é o conjunto de estados iniciais (A,B)-invariante.

6.4.4 Simulação 2

Os dados utilizados para esta simulação são os mesmos utilizados na simulação 1, com exceção do estado inicial:

- $k_0 = [1 \ 66 \ 1 \ 1 \ 6 \ 71 \ 6 \ 6 \ 41 \ 101 \ 41 \ 41]^T$, sendo que $k_0 \in \mathcal{K}$ mas $k_0 \notin \mathcal{K}^\uparrow$.

O semimódulo \mathcal{K}^\uparrow é gerado a partir de \mathcal{K} , sendo que \mathcal{K} é obtido através das restrições, sendo que quando $\mathcal{K}^\uparrow \subsetneq \mathcal{K}$ existem alguns estados que violam alguma restrição durante a evolução do sistema. O objetivo desta simulação é utilizar um estado $k_0 \notin \mathcal{K}^\uparrow$ e ver se é possível através do modelo de tráfego detectar quando uma restrição é violada. A Tabela 6.11 apresenta os resultados encontrados para os quatro grupos de restrições.

Restrições	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$
$s_{11}(k) - x_{11}(k) \leq 0$	0	0	0	0	0
$s_{11}(k) - x_{12}(k) \leq 0$	-21	-16	-16	-16	-16
$s_{21}(k) - x_{21}(k) \leq 0$	-11	-16	-16	-16	-16
$s_{21}(k) - x_{22}(k) \leq 0$	0	-1	0	0	0
$s_{31}(k) - x_{31}(k) \leq 0$	-2	0	-1	-1	-1
$s_{31}(k) - x_{32}(k) \leq 0$	0	0	0	0	0
$x_{11}(k) - s_{12}(k) \leq -44$	-44	-96	-65	-65	-65
$x_{12}(k) - s_{12}(k) \leq -44$	-44	-75	-49	-49	-49
$x_{21}(k) - s_{22}(k) \leq -44$	-44	-80	-49	-49	-49
$x_{22}(k) - s_{22}(k) \leq -44$	-44	-90	-65	-65	-65
$x_{31}(k) - s_{32}(k) \leq -44$	-68	-65	-64	-64	-64
$x_{32}(k) - s_{32}(k) \leq -44$	-70	-65	-65	-65	-65
$s_{11}(k) - s_{21}(k) \leq 5$	-5	-5	0	0	0
$s_{21}(k) - s_{11}(k) \leq 5$	5	5	0	0	0
$s_{21}(k) - s_{31}(k) \leq 40$	-40	-66	-40	-40	-40
$x_{31}(k) - s_{21}(k) \leq 40$	40	66	40	40	40
$s_{12}(k) - s_{11}(k) \leq 70$	44	96	65	65	65
$s_{22}(k) - s_{21}(k) \leq 65$	44	91	65	65	65
$x_{32}(k) - s_{31}(k) \leq 70$	70	65	65	65	65

Tabela 6.11: Resultados encontrados para as restrições – Simulação 2

As restrições dos grupos 1 e 2 são sempre respeitadas, o problema acontece nas restrições definidas nos grupos 3 e 4. Os valores apresentados na Tabela 6.11 para restrição $s_{31}(k) - s_{21}(k) \leq 40$ mostram que o valor máximo definido para o *offset* foi violado. O valor encontrado na simulação foi um *offset* de 66 segundos, sendo que o valor máximo desejado é 40 segundos.

A restrição $s_{22}(k) - s_{21}(k) \leq 65$, que define um valor máximo para o tempo de verde no semáforo S_2 também não é respeitada, e o mesmo acontece com a restrição $s_{12}(k) - s_{11}(k) \leq 70$. Os valores grifados na Tabela 6.11 indicam os valores que violam as restrições.

6.4.5 Simulação 3

Na seção 6.3 foi abordada uma questão importante que deve ser considerada quando se deseja aplicar o controle F em um sistema real: a robustez do controlador em relação aos tempos de viagem d_{ij} . Os valores máximos para d_{ij} que garantem a (A,B)-invariância para o exemplo desenvolvido até o momento são $d_{12} = d_{23} = 40$ segundos e $d_{23} = d_{32} = 45$ segundos, ou seja, caso os veículos gastem mais tempo para atravessar uma via alguma restrição pode ser violada.

O objetivo desta simulação é ver como o sistema se comporta quando se utilizam valores para d_{ij} superiores aos valores máximos. Os dados utilizados são exatamente os mesmos utilizados na simulação 1, inclusive o estado inicial. Os tempos de viagem utilizados são: $d_{12} = d_{23} = 28$ e $d_{12} = d_{23} = 48$. O artifício utilizado para representar o aumento do tempo de viagem d_{ij} é o incremento do número de seções nas vias. O número de seções é definido com base na velocidade de percurso, e aumentar o tempo de viagem é o mesmo que afirmar que a velocidade que os veículos trafegam diminui.

A Tabela 6.12 apresenta os valores encontrados para as restrições quando o tempo de viagem d_{ij} é maior que d_{ij}^{\max} .

Restrições	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$
$x_{11}(k) - s_{12}(k) \leq -44$	-70	-68	-65	-65	-65
$x_{12}(k) - s_{12}(k) \leq -44$	-70	-68	-65	-65	-65
$x_{21}(k) - s_{22}(k) \leq -44$	-65	-68	-65	-65	-65
$x_{22}(k) - s_{22}(k) \leq -44$	-41	-65	-65	-65	-65
$x_{31}(k) - s_{32}(k) \leq -44$	-65	-65	-65	-65	-65
$x_{32}(k) - s_{32}(k) \leq -44$	-65	-65	-65	-65	-65
$s_{11}(k) - s_{21}(k) \leq 5$	-5	0	0	0	0
$s_{21}(k) - s_{11}(k) \leq 5$	5	0	0	0	0
$s_{21}(k) - s_{31}(k) \leq 40$	-40	-40	-37	-37	-37
$x_{31}(k) - s_{21}(k) \leq 40$	40	40	37	37	37
$s_{12}(k) - s_{11}(k) \leq 70$	70	65	65	65	65
$s_{22}(k) - s_{21}(k) \leq 65$	65	65	65	65	65
$x_{32}(k) - s_{31}(k) \leq 70$	65	65	65	65	65
$s_{11}(k) - x_{11}(k) \leq 0$	0	0	0	0	0
$s_{11}(k) - x_{12}(k) \leq 0$	0	0	0	0	0
$s_{21}(k) - x_{21}(k) \leq 0$	0	0	0	0	0
$s_{21}(k) - x_{22}(k) \leq 0$	-24	-3	0	0	0
$s_{31}(k) - x_{31}(k) \leq 0$	0	0	0	0	0
$s_{31}(k) - x_{32}(k) \leq 0$	0	0	0	0	0

Tabela 6.12: Resultados encontrados para as restrições – Simulação 3

É possível perceber que a restrição $x_{22}(k) - s_{22}(k) \leq -44$ pertencente ao grupo 2 é violada, e isto acontece porque o tempo de verde não foi suficiente para absorver um pelotão, e isto aconteceu porque ele se deslocou com uma velocidade inferior à mínima, que é a utilizada para definir a especificação.

As restrições do grupo 3 são sempre respeitadas. Entretanto, o mesmo não acontece com as restrições do grupo 4, onde a restrição $s_{22}(k) - s_{21}(k) \leq 65$ encontra um valor para o tempo de verde superior ao limite especificado.

Na seção 6.3 afirma-se que para o caso onde d_{ij} ultrapasse o valor máximo é possível encontrar uma de duas situações: $\mathcal{K}' \subset \mathcal{K}^\uparrow$ e uma matriz de controle F' , ou $\mathcal{K}' \subseteq \mathcal{K}^\uparrow$ onde não é possível encontrar uma matriz de controle F' . Para obter $\mathcal{K}' \subset \mathcal{K}^\uparrow$ utiliza-se os valores $d_{ij} > d_{ij}^{\max}$ como valores nominais para os tempos de viagem. O comportamento do sistema para este novo conjunto de estados iniciais é analisado na próxima seção.

6.4.6 Simulação 4

A simulação 3 apresentou os resultados obtidos quando se utiliza um tempo de viagem $d_{ij} > d_{ij}^{\max}$, onde se confirma que a especificação não é respeitada. O objetivo desta simulação é encontrar um novo conjunto de estados iniciais $\mathcal{K}' \subset \mathcal{K}^\uparrow$, ou seja, os valores nominais que são utilizados na matriz A que define o sistema são modificados. As restrições não são alteradas, pois justamente se pretende comprovar que é possível encontrar um conjunto de estados iniciais $\mathcal{K}' \subset \mathcal{K}^\uparrow$. O processo para encontrar \mathcal{K}' é o mesmo desenvolvido na seção 6.2.2.

O novo conjunto de estados iniciais calculado é apresentado na matriz da Tabela 6.13. As linhas da matriz são geradoras ou base para o semimódulo $\mathcal{K}' = \text{Im}K'$

205	55	29	27	184	210	35	210	210	194	205	205
270	120	94	74	228	280	100	280	280	238	270	270
205	55	29	27	184	210	35	236	210	194	205	205
205	55	29	27	184	236	35	210	210	194	205	205
210	60	34	32	189	210	40	210	210	189	210	210
275	125	99	79	233	275	105	275	275	233	275	275
210	60	34	32	189	210	40	210	210	189	231	210
210	60	34	32	189	210	40	210	210	189	210	231
170	95	69	72	149	170	72	170	170	149	170	170
235	165	139	116	193	235	142	235	235	193	235	235
170	95	95	72	149	170	72	170	170	149	170	170
170	95	69	72	149	170	98	170	170	149	170	170

Tabela 6.13: Matriz com conjunto de estados iniciais K'

As novas matrizes de controle calculadas são:

$$F_1 = \begin{bmatrix} 129 & 85 & 129 & 129 & 129 & 85 & 129 & 129 & 92 & 45 & 89 & 92 \\ 124 & 80 & 124 & 124 & 129 & 85 & 129 & 129 & 92 & 45 & 89 & 92 \\ 89 & 45 & 89 & 89 & 94 & 50 & 94 & 94 & 132 & 85 & 129 & 132 \end{bmatrix}$$

$$F_2 = \begin{bmatrix} 129 & 85 & 129 & 129 & 127 & 83 & 127 & 127 & 90 & 43 & 87 & 90 \\ 124 & 80 & 124 & 124 & 129 & 85 & 129 & 129 & 92 & 45 & 89 & 92 \\ 87 & 43 & 87 & 87 & 92 & 48 & 92 & 92 & 132 & 85 & 129 & 132 \end{bmatrix}$$

A matriz de controle utilizada no algoritmo de simulação foi F_2 , tendo como estado inicial o seguinte k_0 :

- $k_0 = [1 \ 48 \ 1 \ 1 \ 6 \ 53 \ 6 \ 6 \ 46 \ 90 \ 46 \ 46]^T$, sendo $k_0 = K'(1 : 12, 4) - 26$, ou seja, todos os elementos da quarta coluna de K' descontados de 26 unidades.

As curvas geradas para os *links* l_1 , l_2 e l_3 são apresentadas nas Figuras 6.27, 6.28 e 6.29. Assim como aconteceu na simulação 1 é possível ver a formação de filas nas linhas de parada sempre que um pelotão completo formado em l_1 não consegue atravessar os *links* l_2 e l_3 .

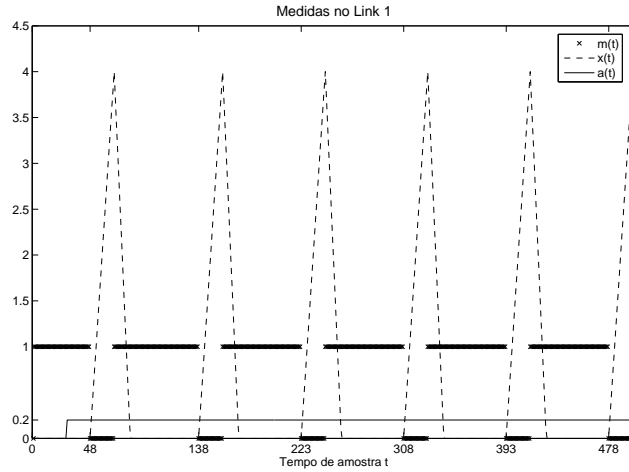


Figura 6.27: Nova evolução das filas $x_1(t)$ em função das chegadas $a_{11}(t)$ e da sinalização $m_1(t)$

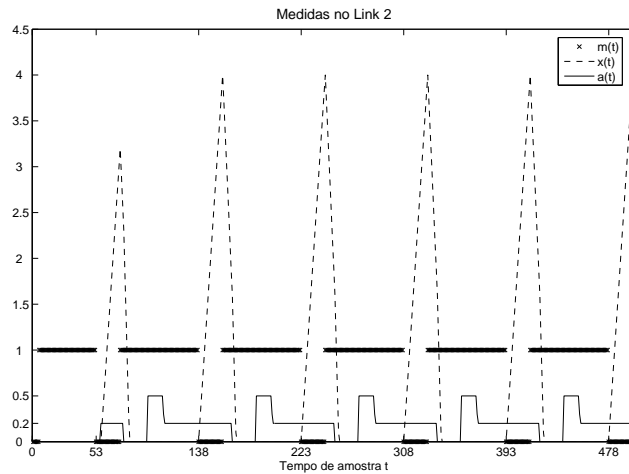


Figura 6.28: Nova evolução das filas $x_2(t)$ em função das chegadas $a_{21}(t)$ e da sinalização $m_2(t)$

As Figuras 6.30, 6.31 e 6.32 apresentam as curvas geradas para os *links* l_4 , l_5 e l_6 . É possível perceber que a chegada de um pelotão em l_6 ocorre um ciclo mais tarde que na simulação 1, e isto se deve justamente ao fato do primeiro pelotão formado em l_4 demorar mais tempo para atravessar a via.

As bandas de verde encontradas nesta simulação são inferiores às calculadas na simulação 1. Na direção $I_1 \rightarrow I_3$ a banda verde é de 29 segundos e na direção $I_3 \rightarrow I_1$ a banda é de 32 segundos. O gráfico da Figura 6.33 apresenta as bandas calculadas.

Os quatro grupos de restrições sempre são garantidos, como pode ser percebido na Tabela 6.14.

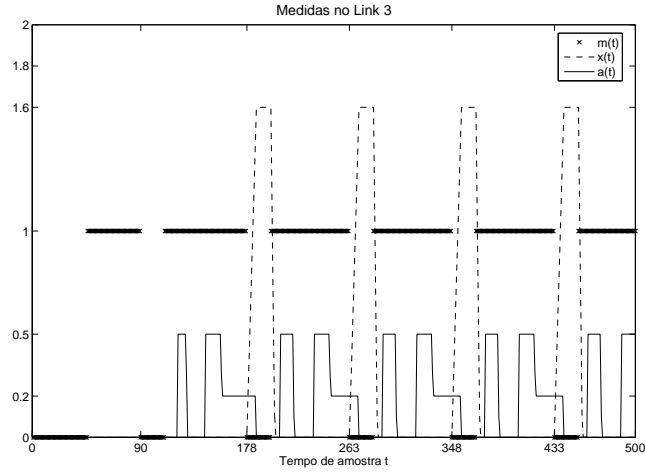


Figura 6.29: Nova evolução das filas $x_3(t)$ em função das chegadas $a_{31}(t)$ e da sinalização $m_3(t)$

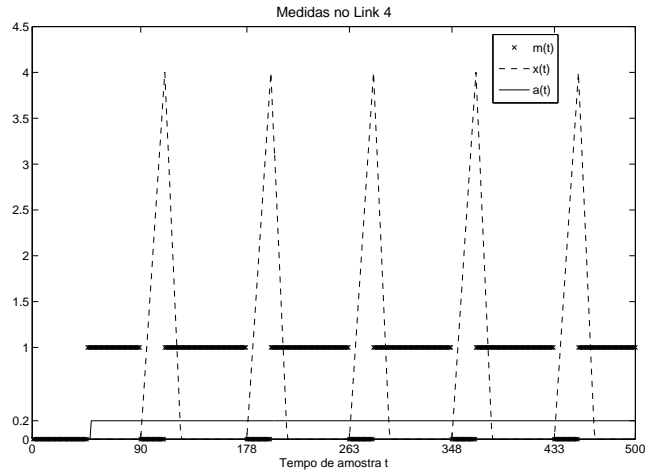


Figura 6.30: Nova evolução das filas $x_4(t)$ em função das chegadas $a_{41}(t)$ e da sinalização $m_4(t)$

Os resultados apresentados comprovam que mesmo para os veículos trafegando em velocidade inferior a utilizada para definir a especificação \mathcal{K} é possível encontrar um conjunto de estados iniciais que vai garantir a (A,B)-invariância do sistema. Mesmo as matrizes K^\dagger e K' tendo as mesmas dimensões, o semimódulo \mathcal{K}' é menor que o semimódulo \mathcal{K}^\dagger , e isto pode ser comprovado utilizando a função `includespan` implementada no *Scilab*.

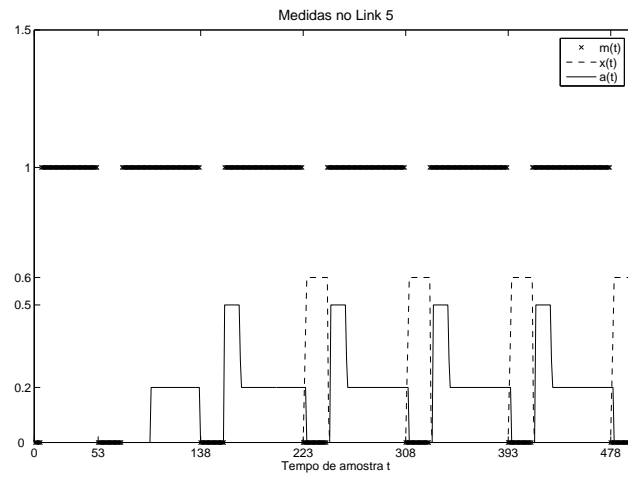


Figura 6.31: Nova evolução das filas $x_5(t)$ em função das chegadas $a_{51}(t)$ e da sinalização $m_5(t)$

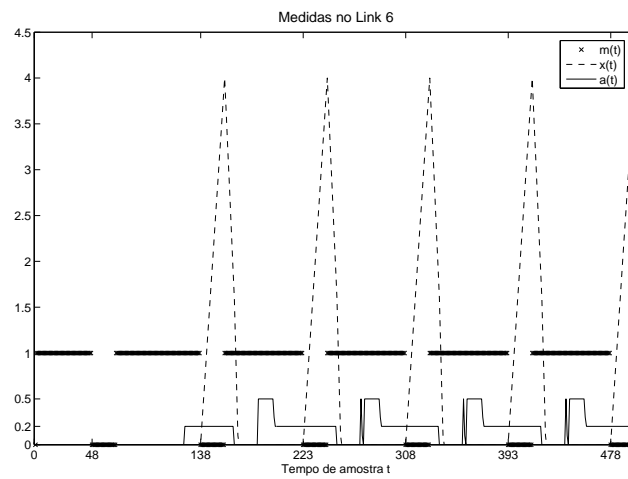


Figura 6.32: Nova evolução das filas $x_6(t)$ em função das chegadas $a_{61}(t)$ e da sinalização $m_6(t)$

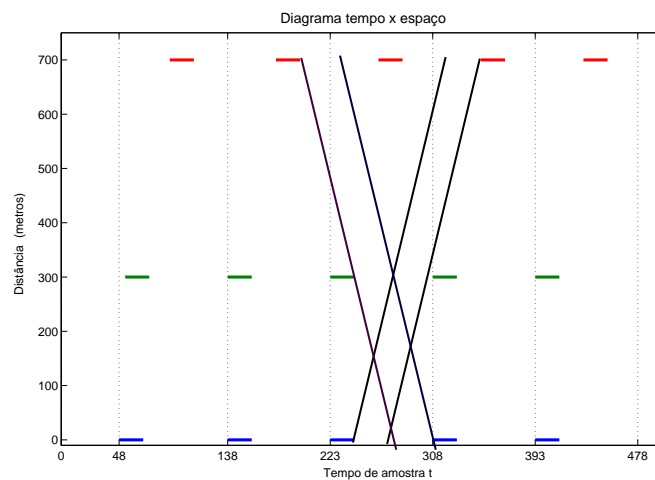


Figura 6.33: Banda de verde gerada – Simulação 4

Restrições	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$
$s_{11}(k) - x_{11}(k) \leq 0$	0	0	0	0	0
$s_{11}(k) - x_{12}(k) \leq 0$	0	0	0	0	0
$s_{21}(k) - x_{21}(k) \leq 0$	0	0	0	0	0
$s_{21}(k) - x_{22}(k) \leq 0$	0	0	0	-3	-3
$s_{31}(k) - x_{31}(k) \leq 0$	0	-11	-8	-8	-8
$s_{31}(k) - x_{32}(k) \leq 0$	0	0	0	0	0
$x_{11}(k) - s_{12}(k) \leq -44$	-47	-70	-65	-65	-65
$x_{12}(k) - s_{12}(k) \leq -44$	-47	-70	-65	-65	-65
$x_{21}(k) - s_{22}(k) \leq -44$	-47	-65	-65	-65	-65
$x_{22}(k) - s_{22}(k) \leq -44$	-47	-65	-65	-62	-62
$x_{31}(k) - s_{32}(k) \leq -44$	-44	-57	-57	-57	-57
$x_{32}(k) - s_{32}(k) \leq -44$	-44	-68	-65	-65	-65
$s_{11}(k) - s_{21}(k) \leq 5$	-5	-5	0	0	0
$s_{21}(k) - s_{11}(k) \leq 5$	5	5	0	0	0
$s_{21}(k) - s_{31}(k) \leq 40$	-40	-37	-40	-40	-40
$x_{31}(k) - s_{21}(k) \leq 40$	40	37	40	40	40
$s_{12}(k) - s_{11}(k) \leq 70$	47	70	65	65	65
$s_{22}(k) - s_{21}(k) \leq 65$	47	65	65	65	65
$x_{32}(k) - s_{31}(k) \leq 70$	44	68	65	65	65

Tabela 6.14: Resultados encontrados para as restrições – Simulação 4

6.5 Conclusão

Ao longo deste capítulo pesquisou-se a aplicação da álgebra max-plus e o uso de semimódulos (A,B)-invariantes para modelar e controlar sistemas de tráfego urbano. Os objetos de estudo foram as vias arteriais semaforizadas, onde existe uma grande necessidade de sincronização.

A metodologia apresentada é definida para um sistema mínimo composto de duas interseções. Conforme mostrado na seção 6.2.1 o modelo pode ser estendido de maneira natural para uma arterial composta de n interseções. O controle do sistema é projetado com base em um grupo de restrições que garantem o bom funcionamento do sistema, e foram definidas de tal maneira que a definição destas restrições é sempre igual, independente do número de interseções da via arterial.

A modelagem das vias arteriais semaforizadas é realizada utilizando uma classe das redes de Petri, os grafos a eventos temporizados (GET), que além de possuir representação gráfica possui uma representação através de equações max-plus lineares. É possível observar que o GET que modela a conexão da via com o semáforo garante a restrição que diz que um pelotão só pode cruzar uma interseção com o semáforo em verde, entretanto, é importante explicitá-la para garantir que o grafo de precedência que representa o conjunto de restrições seja fortemente conexo. O controle obtido aplica uma realimentação de estados na rede de Petri obtida. O método proposto por Katz [2007], que estende o conceito de (A,B)-invariância aos dióides max-plus, é utilizado com sucesso para calcular a matriz que implementa o controle.

O resultado da abordagem proposta neste capítulo é um conjunto de estados iniciais para o sistema que garantem que um conjunto de restrições operacionais e de coordenação, além disso, tempos de verde e vermelho para um sistema em regime permanente também são calculados. A seção 6.2.2 apresenta os resultados encontrados para um exemplo composto de três interseções. Outros resultados podem ser encontrados em Garcia et al. [2007], Garcia et al. [2006a] e Garcia et al. [2006b].

Na seção 6.3 são discutidos alguns pontos importantes para implementar os resultados obtidos no modelo max-plus em um sistema real. O principal ponto é a robustez do controlador, sendo que nesta tese o critério de robustez que interessa está relacionado com a variação dos tempos de viagem d_{ij} em relação aos valores nominais utilizados para encontrar o conjunto de estados iniciais (A,B)-invariante. É apresentado um algoritmo para encontrar os limites d_{ij}^{\max} para os tempos de viagem que garantem a (A,B)-invariância. Além disso, resultados de simulação mostram que a (A,B)-invariância é garantida dentro do intervalo $0 \leq d_{ij} \leq d_{ij}^{\max}$. Existe a perspectiva de provar formalmente esta afirmação.

Um modelo de simulação de tráfego é apresentado com o intuito de utilizar os resultados do modelo max-plus e verificar a possibilidade da aplicação dos resultados obtidos em um sistema real. Os resultados obtidos através de diversas simulações comprovam que é possível definir uma relação entre o modelo desenvolvido através de uma rede de Petri e o modelo de tráfego para simulação. Além disso, utiliza-se a matriz de controle calculada como um controlador que define os instantes de disparo dos sinais de verde no sistema real. Os resultados obtidos permitem a geração de gráficos que ilustram a evolução do tráfego ao longo da via arterial e a verificação da garantia das restrições definidas no modelo max-plus.

O objetivo de propor uma metodologia onde existe uma descrição explícita dos eventos que ocorrem no sistema foi atingido. É possível identificar separadamente os eventos relacionados com a passagem dos pelotões na via e os eventos relacionados com as trocas semafóricas, e isto é possível devido ao fato de construir o modelo global do sistema a partir de modelos individuais da via e do semáforo. Além disso, o controle desejado também é definido de maneira explícita e sistemática, através dos conjuntos de restrições definidos na seção 6.2. A dificuldade encontrada ao utilizar esta metodologia está no cálculo do conjunto de estados iniciais \mathcal{K}^\uparrow , devido a sua complexidade computacional alta, entretanto, soluções para diminuir a complexidade do cálculo de \mathcal{K}^\uparrow continuam sendo investigadas.

Capítulo 7

Conclusões

7.1 Revisão dos Objetivos

Conforme apresentado no Capítulo 1 um aspecto importante do funcionamento do sistema de tráfego urbano veicular urbano é a coordenação semafórica entre interseções adjacentes. Diz-se ainda que os sistemas de tráfego urbano podem ser modelados como sistemas híbridos ou puramente discretos, gerando duas questões que serviram de motivação para a tese: (i) os resultados obtidos para modelagem, análise e controle de sistemas a eventos discretos (SEDs) e híbridos (SHs) podem ser aplicados aos sistemas de tráfego?; e (ii) os formalismos utilizados para SEDs e SHs são mais adequados que os disponíveis para tratar o problema de modelagem e controle de sistemas de tráfego urbano?

Para encontrar respostas para estas perguntas foram definidos dois objetivos para a tese:

1. Desenvolver uma proposta de modelagem para os sistemas de tráfego urbano baseado na teoria de sistemas híbridos, onde os modelos permitem realizar análise do sistema por simulação e verificação de propriedades.
2. Resolver o problema de modelagem e síntese de controle com sincronização semafórica dos sistemas de tráfego urbano através da álgebra max-plus. Especificamente, o problema estudado é o da coordenação semafórica de vias arteriais.

7.2 Contribuições e Resultados da Tese

Ao longo da tese o problema de coordenação semafórica foi estudado sob dois enfoques diferentes: primeiramente os sistemas de tráfego foram analisados como sistemas híbridos; e depois como sistemas a eventos discretos.

O formalismo utilizado para modelar o sistema de tráfego como um sistema híbrido foi o autômato híbrido. Os modelos desenvolvidos expressam tanto as dinâmicas contínuas quanto as dinâmicas

discretas, permitindo assim modelar a influência das dinâmicas discretas dos semáforos na formação das filas, que possuem dinâmicas contínuas.

Na proposta de modelagem apresentada no Capítulo 4 para sistemas de tráfego, utilizando os autômatos híbridos, cada estado do autômato híbrido representa um estágio do semáforo, sendo desenvolvidos modelos individuais para cada cruzamento semaforizado do sistema. A coordenação entre semáforos adjacentes é obtida através da definição das guardas do autômato híbrido. As guardas definem a abertura dos semáforos adjacentes em função do *offset*, que é definido em relação ao primeiro semáforo. Além disso, através das guardas é possível definir os fluxos de chegada das equações das filas dos semáforos a montante com base no tempo de deslocamento na via, garantindo que todas as equações do sistema sejam lineares.

Através dos modelos desenvolvidos é possível simular o sistema e verificar formalmente propriedades. A ferramenta utilizada nesta tese foi o *CheckMate*, um toolbox implementado no *MATLAB*. Através de simulações é possível analisar o comportamento das filas geradas e o funcionamento das guardas. Vale salientar que são desenvolvidos autômatos independentes para cada interseção, e não é necessário obter a composição dos autômatos de cada interseção ao utilizar a ferramenta *CheckMate*, a sincronização se dá através das variáveis globais utilizadas pelas guardas em ambos autômatos.

O objetivo desta tese inclui verificar formalmente propriedades do sistema. Esta etapa do trabalho não pode ser concluída com pleno sucesso devido ao número de variáveis de estados utilizadas nos modelos híbridos. As variáveis de estado são as filas formadas nas interseções e os relógios utilizados para implementar as guardas, por exemplo para modelar as trocas semafóricas. Sendo assim, o número de interseções influencia diretamente na possibilidade de realizar verificação formal de propriedades, devido a complexidade computacional envolvida no cálculo do espaço de estados.

A ferramenta *CheckMate* realiza verificação de sistemas modelados com até 4 variáveis de estado, logo os modelos obtidos para um sistema composto de duas interseções não podem ser verificados. Apesar disso, foram verificadas duas propriedades para um interseção isolada: possibilidade de bloqueio quando a fila atinge um limite máximo e se o tempo de verde utilizado é ocioso para os valores de tráfego definidos.

A propriedade de bloqueio é interessante pois permite identificar se as filas atingem seus valores máximos. A segunda propriedade permite ao engenheiro de tráfego analisar se os tempos semafóricos projetados estão de acordo com as necessidades dos fluxos veiculares existentes. O tempo de verde ocioso em uma direção pode ser transferido para a via transversal, impedindo assim que os veículos formem fila. As propriedades foram traduzidas para especificações em lógica ACTL, e a verificação consiste em garantir se um determinado estado do autômato é alcançado ou não em alguma trajetória.

Através dos resultados obtidos é possível afirmar que as técnicas para sistemas híbridos podem ser aplicados para análise dos sistemas de tráfego; entretanto, o problema da síntese de controle foi abordada apenas para modelos discretos.

Ao estudar o sistema de tráfego como um sistema a eventos discretos as variáveis envolvidas na modelagem são diferentes pois apenas dinâmicas discretas são consideradas. As filas não são

mais modeladas, e sim os semáforos e as vias arteriais. As vias são modeladas explicitamente neste trabalho pois se deseja explicitar os eventos envolvidos, sendo eles as trocas semaforicas e a passagem de pelotões de veículos através das arteriais.

A proposta de modelagem discreta dos sistemas de tráfego apresentada no Capítulo 6 utiliza a álgebra max-plus, e os resultados obtidos são modelos max-plus lineares para o sistema de tráfego. O sistema é modelado através de grafos de eventos temporizados, onde inicialmente são definidos modelos para os semáforos, para as vias e para conexão entre eles. O modelo mínimo é definido para uma arterial composta de duas interseções, e então generalizado para uma arterial com n interseções.

O problema de coordenação semaforica é definido através de um conjunto de restrições que são definidas sobre as transições do grafo de eventos, e as entradas de controle estão nas transições dos semáforos. O método proposto por Katz [2007] define como calcular um semimódulo (A,B)-invariante que contém o conjunto de estados iniciais do sistema onde as restrições são sempre garantidas. Além disso, ele é utilizado para obter uma matriz de realimentação de estados que implementa o controle. Através do semimódulo (A,B)-invariante e da matriz de controle são definidos os tempos semaforicos para o sistema em regime permanente.

A aplicação do método apresentado para arteriais compostas por mais que três interseções é limitada pela complexidade envolvida no cálculo do semimódulo (A,B)-invariante. Entretanto, diversas pesquisas tem sido realizadas para aprimorar os algoritmos do cálculo do máximo semimódulo geometricamente (A,B)-invariante, e a medida que estes resultados sejam disponibilizados será possível aplicar o método para arteriais compostas por mais que três interseções. É importante ressaltar, que mesmo com as limitações computacionais foi possível propor um modelo teórico genérico para n interseções.

Os modelos max-plus lineares e a matriz de controle são calculados utilizando valores nominais para os parâmetros do tráfego. Sabe-se, entretanto, que para implementar o controle em uma aplicação real de tráfego é necessário analisar a robustez da matriz de controle, ou seja, é necessário definir sob quais condições de tráfego a matriz garante que o sistema não viole a especificação desejada. Nesta tese, o critério de robustez estudado está relacionado com a variação do tempo de viagem dos pelotões ao longo da via. Os tempos de viagem variam de acordo com a velocidade que os veículos se deslocam ao longo da via. No Capítulo 6 é apresentado um algoritmo que calcula valores máximos para os tempos de viagem que garantem a robustez da matriz de controle.

A fim de avaliar o comportamento dos modelos discretos obtidos em uma aplicação real de tráfego optou-se por implementar estes modelos em um modelo de tráfego para simulação. No modelo para simulação trabalha-se com as filas e a evolução dos fluxos veiculares ao longo da via. Assim, é necessário estabelecer uma relação entre os eventos e as variáveis do modelo max-plus e o comportamento do tráfego no modelo para simulação. Na seção 6.4 um algoritmo que implementa esta relação é apresentado. Além disso, diversos resultados de simulação são discutidos. Os resultados de simulação permitem avaliar se as restrições especificadas são sempre respeitadas, e os resultados obtidos são satisfatórios. Situações onde se sabe *a priori* que alguma restrição deveria ser violada também foram simuladas, comprovando os resultados esperados sobre a robustez do controle.

Os resultados obtidos ao longo desta tese permitem afirmar que os formalismos e métodos desenvolvidos para sistemas a eventos discretos e híbridos podem ser aplicados aos sistemas de tráfego urbano, respondendo assim a primeira pergunta que motivou esta pesquisa.

A segunda pergunta questiona se os métodos apresentados nesta tese são mais adequados que os métodos clássicos usados em controle de tráfego. Os resultados obtidos não permitem afirmar que um método é melhor ou pior que o outro, mas pode-se afirmar que os métodos desenvolvidos nesta tese são uma abordagem alternativa para o problema da coordenação.

A solução apresentada para a abordagem híbrida do problema permite realizar simulação, e em alguns casos, verificação formal de propriedades. Neste caso é possível identificar uma diferença em relação aos métodos de otimização pois estes métodos não realizam verificação.

Os métodos de otimização que calculam banda máxima de verde encontram uma solução ótima para o problema da coordenação, e o resultado mais conhecido é o algoritmo de Morgan e Little [Morgan e Little, 1964]. Os métodos que otimizam banda de verde calculam os tempos semafóricos de modo *off-line*.

Através dos modelos discretos desenvolvidos no Capítulo 6 realiza-se sincronização e é possível encontrar uma banda de verde. Entretanto, não existe o requisito desta banda ser máxima. A diferença do método apresentado nesta tese é a síntese de controle com realimentação de estados, o que não é possível nos métodos que maximizam banda de verde. A realimentação do controle é útil para fazer ajustes nos tempos semafóricos quando da operação atuada pelo tráfego.

Os resultados referentes à abordagem híbrida do problema de coordenação semafórica foram publicadas em Garcia e Cury [2004a,b]. Os resultados publicados em Garcia et al. [2007, 2006a,b] resumem os resultados da abordagem discreta. Além disso, está sendo preparada uma submissão para uma revista internacional.

7.3 Perspectivas Futuras

Ao longo desta tese os sistemas de tráfego foram estudados como sistemas híbridos e como sistemas a eventos discretos. Em ambas abordagens é possível obter um modelo formal que representa o sistema. Entretanto é possível apontar algumas sugestões de trabalhos futuros que podem melhorar os resultados obtidos até o presente momento:

- modelar outras configurações dos sistemas de tráfego, tanto na abordagem híbrida como na abordagem discreta. No modelos desenvolvidos nesta tese não são consideradas conversões nas interseções e os fluxos veiculares de entrada são constantes;
- investigar o uso de outras ferramentas de verificação formal de propriedades e aplicar aos modelos desenvolvidos. Nesta tese, a verificação formal foi aplicada apenas para uma interseção isolada;

- os resultados de simulação obtidos com os modelos discretos mostram que sendo d_{ij}^{max} o tempo de viagem máximo d_{ij}^{max} que garante (A,B)-invariância, qualquer valor dentro do intervalo $0 \leq d_{ij} \leq d_{ij}^{max}$ garante a (A,B)-invariância. Deseja-se provar formalmente esta hipótese;
- investigar como calcular as entradas de controle u quando o semimódulo \mathcal{K}^\dagger não é algebricamente (A,B)-invariante, ou seja, quando não é possível encontrar uma matriz de realimentação F ;
- investigar como o engenheiro de tráfego pode utilizar os resultados desta tese, ou seja, definir como usar as informações em uma ferramenta de apoio a decisão.

Apêndice A

Terminologia de Tráfego Urbano

Para melhor entendimento desta tese é necessário introduzir alguns termos que são correntes na literatura de tráfego urbano. As definições aqui apresentadas em ordem alfabética foram baseadas nos trabalhos de Papageorgiou et al. [2003], de Brito [1998], Cervantes [2005], Carlson [2006] e DEN [1984].

Aproximação ou *link*: Denominação dada para uma via que chega em uma interseção.

Capacidade da interseção: É o fluxo total máximo de veículos que pode passar através da interseção durante um intervalo de tempo.

Concentração: É o número de veículos ocupando uma unidade de comprimento de uma pista num instante de tempo. Quando expressa em veh/km é chamada densidade.

Coordenação e sincronização: A coordenação consiste na obtenção de defasagem para os semáforos dados porcentagens de verde de cada interseção e tempo de ciclo comum. A sincronização é a obtenção simultânea de todas as variáveis de controle, defasagem, porcentagens de verde e tempo de ciclo comuns. Em alguns trabalhos os termos coordenação e sincronização são usados indistintamente, como em Carlson [2006].

Estágio: Parte de um ciclo durante o qual um grupo de movimentos têm permissão de passagem. A Figura A.1 mostra um diagrama de estágios para a interseção da Figura A.2. O ciclo é dividido em três estágios e são definidas quatro fases, uma para cada movimento. Um estágio também é definido como o conjunto de fases de todas as aproximações de uma interseção em determinado instante de tempo.

Fase: Geralmente a seqüência de indicação de cores em um semáforo é verde, amarelo, vermelho e novamente verde. Esta seqüência aplicada a um ou mais movimentos é denominada fase.

Os exemplos estudados nesta tese utilizam apenas a seqüência verde, vermelho e novamente verde. O amarelo é desconsiderado. No exemplo apresentado na Figura A.1 foi definida uma fase para cada movimento, entretanto, existem casos onde é possível definir uma fase para um conjunto de movimentos (ver exemplos do Capítulo 4).

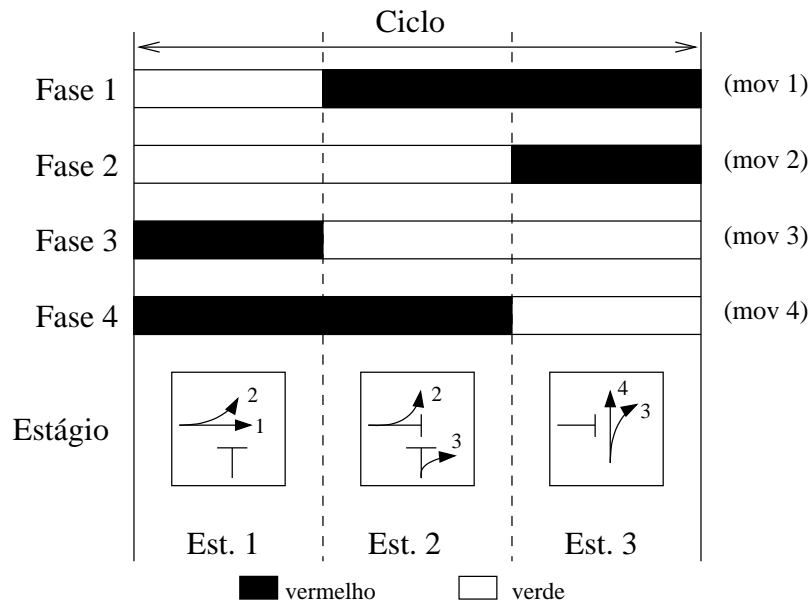


Figura A.1: Diagrama de estágios – exemplo

Fila: É um grupo de veículos estacionário na linha de parada de uma via.

Fluxo de saturação: Taxa máxima de veículos em relação ao tempo, que seriam descarregados de uma via se esta contasse com uma fila de tamanho infinito e indicação semafórica verde.

Fluxo de tráfego: Número de veículos que passam por um ponto durante um período de tempo especificado. Quando expresso em veh/h é chamado de volume.

Grau de saturação: É definido como a relação entre o número médio de veículos que chegam ao cruzamento durante o ciclo através de uma faixa e o número de veículos que podem ser atendidos pelo cruzamento através desta faixa durante um ciclo, sendo dado por:

$$x = \frac{q}{s} \cdot \frac{C}{g} = y \frac{C}{g}$$

onde q é o fluxo, C é o comprimento de um ciclo, s é o fluxo de saturação, g é o tempo efetivo de verde e y a taxa de ocupação da via. Quando o grau de saturação é maior que um significa que chegam mais veículos do que podem ser atendidos naquela faixa. Quando esta situação dura muito tempo as filas crescem e o sistema torna-se saturado.

Interseção ou cruzamento: É o local onde duas ou mais vias se cruzam, criando um conflito entre os sentidos de circulação de veículos.

Largura de banda de verde: É a distância temporal entre a linha imaginária do percurso do primeiro e do último veículo que passam desimpedidos por todos os semáforos de uma via arterial. A banda de verde pode ocorrer em ambos os sentidos da via, e as larguras não são necessariamente iguais. A largura da banda de verde é definida com base em parâmetros da via e por meio da coordenação apropriada dos semáforos.

Movimentos: Possíveis deslocamentos de veículos numa interseção. A Figura A.2 apresenta um conjunto de 4 movimentos para uma interseção onde as vias são de mão única.

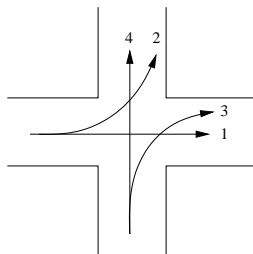


Figura A.2: Conjunto de movimentos

Movimentos conflitantes: Dois ou mais movimentos são conflitantes entre si quando se cruzam numa interseção. É possível observar na Figura A.2 que os movimentos 1 e 4 são conflitantes, ele não podem acontecer ao mesmo tempo pois existe perigo de colisão.

Ocupação: A taxa de ocupação y de uma aproximação é a razão entre o fluxo de veículos (q) em uma via e o fluxo de saturação (s), isto é, $y = q/s$.

Offset ou defasagem: Diferença de tempo entre o início de estágios pré-determinados de duas interseções semaforizadas consecutivas. É positiva se o estágio principal da interseção a jusante inicia após o estágio principal da interseção a montante, e negativa caso contrário. O *offset* se aplica na sincronização entre interseções que são operadas de forma coordenada como um sistema.

Pelotão: Um grupo de veículos que atravessa uma via, sem que ocorra dispersão.

Semáforo: Dispositivo de controle de tráfego que alterna o direito de passagem de veículos e pedestres em interseções mediante a utilização de indicadores luminosos. Este sistema de controle organiza de forma cíclica e seqüencial a passagem de veículos e pedestres em uma interseção.

Split: O *split* de um ciclo é a forma como o mesmo está dividido entre as estágios, ou seja, é o conjunto de frações de verde do ciclo atribuídas a cada estágio.

Taxa de chegada: É a taxa segundo a qual os veículos chegam em uma determinada faixa de uma pista. Esta taxa pode ser considerada conhecida e constante, ou ainda, medida através de detectores.

Taxa de descarga: É a máxima taxa possível em que uma fila formada em uma faixa da pista é descarregada; esta taxa é igual ao fluxo de saturação enquanto existir fila a ser descarregada.

Tempo de ciclo: Também chamado de ciclo. Corresponde ao tempo do início do estágio principal de uma interseção até o instante em que ele se inicia novamente, tendo ocorrido outros estágios neste período. Também é o tempo total para a completa seqüência de sinalização numa interseção.

Tempo de deslocamento: É o tempo que um veículo leva para alcançar um semáforo partindo de outro. Também denominado tempo de viagem.

Velocidade de cruzeiro: É considerada como a velocidade que os veículos atingem quando percorrem uma determinada distância sem que ocorram interrupções. Esta velocidade depende das condições geométricas das vias e das condições de tráfego.

Vermelho de segurança: É o intervalo de tempo dentro de um ciclo em que nenhum conjunto de movimentos tem o direito de passagem pela interseção, por questões de segurança.

Vias, pistas e faixas: Uma faixa de trânsito é o espaço determinado para o fluxo de veículos em um sentido único de fluxo. O termo pista é utilizado para identificar um conjunto de faixas de trânsito. Uma via pode ser composta de diversas pistas, separadas por canteiros centrais. Cada pista é composta por uma ou várias faixas. A Figura A.3 ilustra a diferença entre estes termos.

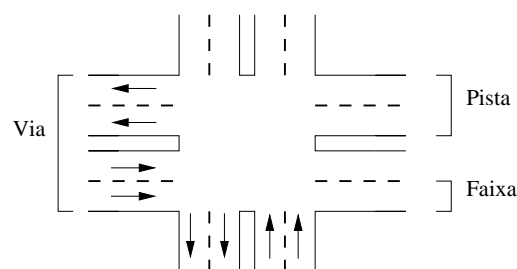


Figura A.3: Definição de via, pista e faixa

Além disso, uma via é chamada arterial quando a velocidade média é alta, caracterizando tráfego de médio e longo percurso. Ela tem prioridade nos cruzamentos e não é permitido estacionamento de veículos.

Referências Bibliográficas

- Alla, H. e David, R. “On Hybrid Petri Nets”. *Journal of Discrete Event Dynamic Systems*, 11(1–2): 9–40, 2001.
- Alur, R., Courcoubetis, C., e Dill, D. “Model-checking for real-time systems”. In *Proceedings of the 5th IEEE Symposium on Logic in Computer Science*, pp. 414–425, 1990.
- Alur, R., Courcoubetis, C., Henzinger, T. A., e Ho, P.-H. “Hybrid Automata: an algorithmic approach to the specification and verification of hybrid systems”. In Grossman et al. [1993], pp. 209–229. ISBN 3-540-57318-6.
- Alur, R., Henzinger, T. A., Lafferriere, G., e Pappas, G. J. “Discrete abstractions of Hybrid Systems”. *Proceedings of the IEEE*, 88(7):971–984, julho de 2000.
- Alur, R., Dang, T., e Ivancic, F. “Predicate Abstraction for Reachability Analysis of Hybrid Systems”. *ACM Transactions on Embedded Computing Systems*, 5(1):152–199, february 2006.
- Alur, R. e Dill, D. L. “A Theory of Timed Automata”. *Theoretical Computer Science*, 126(2):183–235, 1994.
- Alur, R., Henzinger, T. A., e Ho, P.-H. “Automatic symbolic verification of embedded systems”. *IEEE Transactions on Software Engineering*, 22(3):181–201, março de 1996a.
- Alur, R., Henzinger, T. A., e Sontag, E. D., editors. *Hybrid Systems III: Verification and Control*, volume 1066 de *Lecture Notes in Computer Science*. Springer-Verlag, 1996b. ISBN 3-540-61155-X.
- Amari, S. *Commande des Graphes d’Événements Temporisés sous Contraintes Temporelles*. PhD thesis, Université de Nantes – École Doctorale, Nantes, France, 2005.
- Antsaklis, P., Kohn, W., Nerode, A., e Sastry, S., editors. *Hybrid Systems II*, volume 999 de *Lecture Notes in Computer Science*. Springer-Verlag, 1995. ISBN 3-540-60472-3.
- Antsaklis, P., Stiver, J. A., e Lemmon, M. D. “Hybrid systems modeling and autonomous control systems”. In Grossman et al. [1993], pp. 366–392. ISBN 3-540-57318-6.
- Antsaklis, P. J., Kohn, W., Lemmon, M., Nerode, A., e Sastry, S., editors. *Hybrid Systems V*, volume 1567 de *Lecture Notes in Computer Science*. Springer-Verlag, 1999. ISBN 3-540-65643-X.

- Antsaklis, P. J. “Special issue hybrid systems: theory and applications”. *Proceedings of the IEEE*, 88 (7):879–887, julho de 2000.
- Baccelli, F., Cohen, G., Olsder, G. J., e Quadrat, J.-P. *Synchronization and Linearity*. Wiley, 1992.
- Back, A., Guckenheimer, J., e Myers, M. “A dynamical simulation facility for hybrid systems”. In Grossman et al. [1993], pp. 255–267. ISBN 3-540-57318-6.
- Bengtsson, J., Larsen, K. G., Larsson, F., Pettersson, P., Wang, Y., e Weise, C. “New generation of UPPAAL”. In *In Proceedings of International Workshop on Software Tools for Technology Transfer*, julho de 1998.
- Benveniste, A. “Compositional and uniform modelling of hybrid systems”. In Alur et al. [1996b], pp. 41–51. ISBN 3-540-61155-X.
- Blyth, T. e Janowitz, M. *Residuation Theory*. Pergamon Press, Oxford, 1972.
- Boillot, F., Blosseville, J. M., Lesort, J. B., Motyka, V., Papageorgiou, M., e Sellam, S. “Optimal Signal Control of Urban Traffic Networks”. In *Proceedings of the 6th IEEE International Conference on Road Traffic Monitoring and Control*, pp. 75–79, 1992.
- Borriello, G. e Walkup, E. “A General Linear Max-Plus Solution Technique”. In Gunawardena, J., editor, *Idempotency (Bristol, 1994)*, pp. 406–415. Cambridge University Press, Cambridge, 1998.
- Brandin, B. A. “The real-time supervisory control of an experimental manufacturing cell”. *IEEE Transactions on Robotics and Automation*, 12(1):1–14, fevereiro de 1996.
- Branicky, M. S. “General hybrid dynamical systems: modeling, analysis and control”. In Alur et al. [1996b], pp. 186–200. ISBN 3-540-61155-X.
- Branicky, M. S., Borkar, V. S., e Mitter, S. K. “Stability of switched and hybrid systems”. In *Proceedings of the 33rd IEEE Conference on Decision and Control*, pp. 3498–3503, Lake Buena Vista, FL, dezembro de 1994.
- Brockett, R. W. “Hybrid models for motion control systems”. In Trentelman, H. L. e Willems, J. C., editors, *Perspectives in Control*, pp. 29–54, Birkhauser, Boston, 1993.
- Butkovic, P. e Hegedüs, G. “An Elimination Method for finding all Solutions of the System of Linear Equations over an Extremal Algebra”. *Ekonomicko-matematicky Obzor*, 20(2):203–215, 1984.
- Carlson, R. C. *Aplicação de Maximização de Largura de Banda no Controle de Tráfego Urbano em Tempo-Real*. Dissertação (mestrado), Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Santa Catarina, Florianópolis, SC, 2006.
- Cassandras, C. G. e Lafortune, S. *Introduction to Discrete Event Systems*. Kluwer, 1999.
- Cellier, F. E. *Combined continuous/discrete system dimulation by use of digital computers: techniques and tools*. Tese (doutorado), Swiss Federal Institute of Technology, Zurich, Switzerland, 1979.

- Cervantes, S. G. *Um Algoritmo Descentralizado para Controle de Tráfego Urbano em Tempo Real*. Tese (doutorado), Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Santa Catarina, Florianópolis, SC, 2005.
- Chutinan, A. *Hybrid system verification using discrete model approximations*. Tese (doutorado), Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, USA, maio de 1999.
- Chutinan, A. e Krogh, B. H. “Computing polyhedral approximations to dynamic flow pipes”. In *Proceedings of 37th IEEE Conference on Decision Control*, 1998.
- Chutinan, A. e Krogh, B. H. “Verification of polyhedral-invariant hybrid automata using polygonal Flow Pipe approximations”. In Vaandrager, F. W. e van Schuppen, J. H., editors, *Hybrid Systems: Computation and Control*, volume 1569 de *Lecture Notes in Computer Science*, pp. 76–90. Springer-Verlag, 1999.
- Chutinan, A. e Krogh, B. H. “Infinite-state transition system verification using approximative quotient transition systems”. *IEEE Transactions on Automatic Control*, 46(9):1401–1410, setembro de 2001.
- Chutinan, A. e Krogh, B. H. “Computational techniques for hybrid system verification”. *IEEE Transactions on Automatic Control*, 48(1):64–75, janeiro de 2003.
- Cuningham-Green, R. A. e Butkovic, P. “The Equation $A \otimes x = B \otimes y$ over $(\max, +)$ ”. *Theoretical Computer Science*, 293(1):3–12, 2003.
- Cury, J. E. R. e Krogh, B. H. “Synthesizing supervisory controllers for hybrid systems”. *Journal of the Society of Instrument and Control Engineers*, 38(3):161–168, março de 1999.
- Cury, J. E. R., Krogh, B. H., e Niinomi, T. “Synthesis of supervisory controllers for hybrid systems based on approximating automata”. *IEEE Transactions on Automatic Control*, 43(4):564–568, abril de 1998.
- Dang, T. *Vérification et synthèse des systèmes hybrides*. Tese (doutorado), Institut National Polytechnique de Grenoble, VERIMAG, Grenoble, França, outubro de 2000.
- Dang, T. e Maler, O. “Reachability analysis via face lifting”. In *Proceedings of Hybrid Systems: Computation and Control (HSCC’98)*, volume 1386 de *Lecture Notes in Computer Science*, pp. 96–109, Berkeley, California, USA, abril de 1998. Springer-Verlag.
- David, R. e Alla, H. “On hybrid Petri nets”. *Discrete Event Dynamic Systems: Theory and Applications*, 11:9–40, 2001.
- de Brito, R. M. “Desenvolvimento de um simulador para análise e projeto de sistemas de controle de tráfego em malha fechada”. Dissertação (mestrado), Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Santa Catarina, Florianópolis, SC, 1998.
- de Vries, R., Schutter, B. D., e Moor, B. D. “On Max-algebraic Models for Transportation Networks”. In *Proceedings of WODES’98*, pp. 457–462, Cagliari, Italy, 1998.

- Demongodin, I. e Koussoulas, N. T. “Differential Petri nets: representing continuous systems in a discrete-event world”. *IEEE Transactions on Automatic Control, Special Issue on Hybrid Control Systems*, 43(4):573–579, abril de 1998.
- Manual de Semáforos*. DENATRAN, Brasília, 1984.
- Diakaki, C. M., Dinopoulou, V., Aboudolas, K., Papageorgiou, M., Ben-Shabat, E., Seider, E., e Leibov, A. “Extensions and new applications of the traffic signal control strategy TUC”. *Transportation Research Record*, (1856):202–216, 2003.
- DiCesare, F., Kulp, P., Gile, K., e List, G. “The applications of Petri nets to the modeling, analysis and control of intelligent urban traffic networks”. *Lecture Notes in Computer Science*, 815, 1994.
- Dinopoulou, V., Diakaki, C., e Papageorgiou, M. “Simulation investigations of the traffic-responsive urban control strategy TUC”. In *Proc. of 3rd IEEE Conference on Intelligent Transportation Systems*, pp. 458–463, Dearborn, Michigan, outubro de 2000.
- Ezzine, J. e Haddad, A. H. “Controllability and observability of hybrid systems”. *International Journal of Control Proceedings*, 49(6):2045–2055, 1989.
- Fahrland, D. A. “Combined discrete event continuous systems simulation”. *Simulation*, 14(2):61–72, 1970.
- Farges, J.-L., Henri, J.-J., e Tufal, J. “The PRODYN real-time traffic algorithm”. In *Proceedings of 4th IFAC Symposium Transportation Systems*, pp. 307–312, 1983.
- Febbraro, A. D., Giglio, D., e Sacco, N. “Modular representation of urban traffic systems based on hybrid Petri nets”. In *Proceedings of the IEEE Intelligent Transportation Systems Conference*, pp. 866–871, Oakland, CA, USA, agosto de 2001.
- Febbraro, A. D., Giglio, D., e Sacco, N. “On controlling privileged vehicles by means of coordinated traffic lights”. In *Proceedings of the IEEE Intelligent Transportation Systems Conference*, volume 2, pp. 1318–1323, Shanghai, China, outubro de 2003.
- Febbraro, A. D. e Sacco, N. “On modelling urban transportation networks via hybrid Petri nets”. *Control Engineering Practice*, 12(10):1225–1239, outubro de 2004.
- Frehse, G. “PHAVer: Algorithmic Verification of Hybrid Systems Past HyTech”. In Morari, M. e Thiele, L., editors, *Proceedings of Hybrid Systems: Computation and Control (HSCC 2005)*, volume 3414 de *Lecture Notes in Computer Science*, pp. 258–273, Zurich, Switzerland, março de 2005. Springer-Verlag.
- Gallego, J.-L., Farges, J.-L., e Henry, J.-J. “Design by Petri nets of an intersection signal controller”. *Transportation Research C*, 4(4):231–248, 1996.
- Garcia, T. R. e Cury, J. E. R. “Modelagem e verificação de sistemas de tráfego urbano através de autômatos híbridos”. In *Proceedings XV Congresso Brasileiro de Automática (CBA 2004)*, pp. 704–709, Gramado, RS, setembro de 2004a.

- Garcia, T. R. e Cury, J. E. R. “Uma abordagem de sistemas híbridos para modelagem e verificação de sistemas de tráfego urbano”. In *Proceedings XIII Panamerican Conference of Traffic and Transportation Engineering (PANAM 2004)*, pp. 1–12, Albany, NY (EUA), setembro de 2004b.
- Garcia, T. R., Cury, J. E. R., e Junior, W. K. “Modelagem e controle de vias arteriais urbanas através da álgebra max-plus”. In *Proceedings XVI Congresso Brasileiro de Automática (CBA 2006)*, pp. 983–988, Salvador, BA, oct 2006a.
- Garcia, T. R., Cury, J. E. R., e Junior, W. K. “Modelagem e Projeto de Coordenação Semafórica Através da Álgebra Max-plus”. In *Proceedings of XIX Congresso Nacional da ANPET*, Brasília, DF, 2006b.
- Garcia, T. R., Cury, J. E. R., Junior, W. K., e Demongodin, I. “Traffic Light Coordination of Urban Corridors Using Max-Plus Algebra”. In *Proceedings of International Modeling and Simulation Multiconference 2007 (IMSM 2007)*, pp. 255–260, Buenos Aires, Argentina, feb 2007.
- Gartner, N. H. “OPAC: A demand-responsive strategy for traffic signal control”. *Transportation Research Record*, 906:75–84, 1983.
- Gartner, N. H., Assman, S. F., Lasaga, F., e Hou, D. L. “MULTIBAND - A Variable-Bandwidth Arterial Progression Scheme”. *Transportation Research Record*, 1(1287):212–222, 1990.
- Gaubert, S. e Gunawardena, J. “The Duality Theorem for Min-Max Functions”. *Comptes Rendus de l'Académie des Sciences*, 326(1):43–48, 1998.
- Gaubert, S. *Théorie des Systèmes linéaires dans les Dioïdes*. PhD thesis, École des Mines de Paris, 1992.
- Gazis, D. C. *Traffic theory*. Kluwer Academic Publishers, 2002.
- Gerlough, D. L. e Huber, M. J. *Traffic Flow Theory: A Monograph*. Transportation Research Board, National Research Council, Washington, 1975.
- Girard, A. R., de Souza, J. B., Misener, J. A., e Hedrick, J. K. “A control architecture for integrated cooperative cruise control and collision warning systems”. In *Proceedings of the Control and Decision Conference*, Florida, dezembro de 2001.
- Godbole, D. N. e Lygeros, J. “Longitudinal control of the Lead Car of a Platoon”. *IEEE Transactions on Vehicular Technology*, 43(4):1125–1135, novembro de 1994.
- Grossman, R. L., Nerode, A., Ravn, A. P., e Rischel, H., editors. *Hybrid Systems*, volume 736 de *Lecture Notes in Computer Science*. Springer-Verlag, New York, dezembro de 1993. ISBN 3-540-57318-6.
- Heidergott, B., Olsder, G. J., e van der Woude, J. *Max Plus at Work*. Princeton University Press, 2006.
- Henzinger, T. A. “Hybrid automata with finite bisimulations”. In Vaandrager, F. e van Schuppen, J., editors, *Proceedings ICALP'95*, volume 944 de *Lecture Notes in Computer Science*, pp. 324–335. Springer-Verlag, 1995.

- Henzinger, T. A., Ho, P.-H., e Wong-Toi, H. "HYTECH: A Model Checker for Hybrid Systems". *Software Tools for Technology Transfer*, 1:110–122, 1997.
- Henzinger, T. A. e Sastry, S., editors. *Proceedings of Hybrid Systems: Computation and Control (HSCC'98)*, volume 1386 de *Lecture Notes in Computer Science*, Berkeley, California, USA, abril de 1998. Springer-Verlag. ISBN 3-540-64358-3.
- Horowitz, R. e Varaiya, P. "Control design of an automated highway system". *Proceedings of the IEEE, Special Issue on Hybrid Systems: Theory and Applications*, 88(7):913–925, julho de 2000.
- Hunt, P. B., Robertson, D. I., Bretherton, R. D., e Winton, R. I. "SCOOT: a traffic responsive method of co-ordinating signals". Report 1024, TRL Laboratory, 1981.
- Hunt, P. B., Robertson, D. L., e Bretherton, R. D. "The SCOOT on-line traffic signal optimization technique". *Traffic Engineering and Control*, 23:190–192, 1982.
- Johnson, T. L. "Analytic models of multistage processes". In *Proceedings of the 20th IEEE Conference on Decision and Control*, San Diego, California, 1981.
- Joseph, M., editor. *Formal Techniques in Real-Time and Fault-Tolerant Systems*, volume 331 de *Lecture Notes in Computer Science*, Warwick, UK, setembro de 1988. Springer-Verlag.
- Kang, Y. *Delay, stop and queue estimation for uniform and random traffic arrivals at fixed-time signalized intersections*. PhD thesis, Faculty of the Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 2000.
- Katz, R. D. *Problemas de Alcanzabilidad e Invariancia en el Álgebra Max-plus*. PhD thesis, National University of Rosario, 2003.
- Katz, R. D. "Max-plus (A,B)-Invariant Spaces and Control of Timed Discrete Event Systems". *IEEE Transactions on Automatic Control*, 52(2):229–241, feb 2007.
- Kowalewski, S. "Introduction to the analysis and verification of hybrid systems". In *Proceedings of Modelling, Analysis and Design of Hybrid Systems*, volume 279, pp. 153–172. Springer-Verlag, 2002.
- Lafferriere, G., Pappas, G., e Yovine, S. "Reachability computation for linear systems". In *Proceedings of the 14th IFAC World Congress*, volume E, pp. 7–12, 1999.
- Lafferriere, G. "Discontinuous stabilizing feedback using partially defined lyapunov functions". In *Proceedings of the IEEE Conference on Decision and Control*, pp. 3487–3491, dezembro de 1994.
- Leal, A. e Cury, J. E. R. "Modular supervision of hybrid systems: a DES approach". In *Proceedings of 7th IFAC Workshop on Discrete Event Systems (WODES'04)*, Reims, França, setembro de 2004a.
- Leal, A. e Cury, J. E. R. "On the existence of optimal solutions for the modular supervisory control of hybrid systems". In *Proceedings of IEEE Conference on Decision and Control*, Bahamas, dezembro de 2004b.

- Leal, A. B. *Controle supervisorio modular de sistemas híbridos*. Tese (doutorado), Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Santa Catarina, Florianópolis, SC, 2005.
- Lei, J. e Ozguner, U. “Decentralized Hybrid Intersection Control”. In *Proceedings of The 40th Conference of Decision and Control*, pp. 1237–1242, Orlando, Florida USA, 2001.
- Little, J. D. C. “The Synchronization of Traffic Signals by Mixed-Integer Linear Programming”. *Operations Research*, 14:568–594, 1966.
- Little, J. D. C., Kelson, M. D., e Gartner, N. H. “MAXBAND: A Program for Setting Signals on Arteries and Triangular Networks”. *Transportation Research Record*, 1(795):40–46, 1981.
- Livadas, C. *Formal verification of safety-critical hybrid systems*. Tese (mestrado), Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, setembro de 1997.
- Livadas, C., Lygeros, J., e Lynch, N. A. “High-Level Modeling and Analysis of the Traffic Alert and Collision Avoidance System (TCAS)”. *Proceedings of the IEEE, Special Issue on Hybrid Systems: Theory and Applications*, 88(7):926–948, julho de 2000.
- Livadas, C. e Lynch, N. A. “Formal verification of safety-critical hybrid systems”. In Henzinger, T. A. e Sastry, S., editors, *Hybrid Systems: Computation and Control*, volume 1386 de *Lecture Notes in Computer Science*, pp. 253–272. Springer-Verlag, 1998.
- Lowrie, P. R. “The Sydney co-ordinated adaptive traffic system: principles, methodology, algorithms”. In *Proceedings of the International Conference on Road Traffic Signalling*, pp. 67–70, London, 1982. Institution of Electrical Engineers.
- Lynch, N. A. e Krogh, B. H., editors. *Proceedings of Hybrid Systems: Computation and Control (HSCC 2000)*, volume 1790 de *Lecture Notes in Computer Science*, Pittsburgh, PA, USA, março de 2000. Springer-Verlag. ISBN 3-540-67259-1.
- Maia, C. A. *Identificação e Controle de Sistemas a Eventos Discretos em Álgebra (max, +)*. Tese (doutorado), Programa de Pós-Graduação da Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas, Campinas, SP, 2003.
- Maia, C. A., Lüders, R., Mendes, R. S., e Hardouin, L. “Estratégias de Controle por Modelo de Referência de Sistemas a Eventos Discretos Max-Plus Lineares”. *Controle e Automação – Revista da Sociedade Brasileira de Automática*, 16(3):263–278, 2005a.
- Maia, C. A., Mendes, R. S., e Hardouin, L. “Identificação de Sistemas a Eventos Discretos Maxplus Lineares”. *Controle e Automação – Revista da Sociedade Brasileira de Automática*, 16(4):407–416, 2005b.
- Maler, O., Manna, Z., e Pnueli, A. “From Timed to Hybrid Systems”. In *Proceedings of the REX Workshop on Real-Time: Theory in Practice*, volume 600 de *Lecture Notes in Computer Science*, pp. 474–484. Springer-Verlag, 1992.

- Manna, Z. e Pnueli, A. *Temporal verification of reactive systems: Safety*. Springer-Verlag, 1995.
- Manna, Z., Bjorner, N., Browne, A., Chang, E., Colon, M., Kapur, A., Sipma, H., e Uribe, T. “STeP: Deductive-algorithmic verification of reactive and real-time systems”. In *Proceedings of the 8th International Conference on Computer-Aided Verification*, Lecture Notes in Computer Science. Springer-Verlag, 1996.
- Manna, Z. e Pnueli, A. *The Temporal Logic of Reactive and Concurrent Systems*. Springer-Verlag, 1992.
- Max Plus. “Max-plus Toolbox for Scilab”, 1998. <http://www-rocq.inria.fr/scilab>.
- McMillan, K. L. *Symbolic Model Checking: An approach to the state explosion problem*. Kluwer Academic Publishers, 1993.
- Miranda, M. V. C. *Contribuição ao uso de métodos formais no estudo de circuitos de eletrônica de potência*. Tese (doutorado), Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Campina Grande, Campina Grande, PB, 2003.
- Morgan, J. T. e Little, J. D. C. “Synchronizing Traffic Signals for Maximal Bandwidth”. *Operations Research*, 12(6):896–912, 1964.
- Murata, T. “Petri nets: properties, analysis and applications”. *Proceedings of IEEE*, 77(4):541–580, 1989.
- Nerode, A. e Kohn, W. “Models for hybrid systems: Automata, topologies, controllability, observability”. In Grossman, R. L., Nerode, A., Ravn, A. P., e Rischel, H., editors, *Hybrid Systems*, volume 736 de *Lecture Notes in Computer Science*, pp. 317–356. Springer-Verlag, New York, NY, dezembro de 1993.
- Nicolin, X., Olivero, A., Sifakis, J., e Yovine, S. “An approach to the description and analysis of hybrid automata”. In Grossman et al. [1993], pp. 149–178. ISBN 3-540-57318-6.
- Papageorgiou, M., Diakaki, C., Dinopoulou, V., Kotsialos, A., e Wang, Y. “Review of road traffic control strategies”. *Proceedings of the IEEE*, 91(12):2043–2067, dezembro de 2003.
- Pavlidis, T. “Stability of systems described by differential equations containing impulses”. *IEEE Transactions on Automatic Control*, 12(1):43–57, 1967.
- Payne, H. J. “Models of freeway traffic and control”. In *Mathematical Models of Public Systems*, volume 1 de *Simulation Council Proceedings Series*, pp. 51–61. G. A. Bekey, La Jolla, California, 1971.
- Pepyne, D. L. e Cassandras, C. G. “Optimal control of hybrid systems in manufacturing”. *Proceedings of the IEEE, Special Issue on Hybrid Systems: Theory and Applications*, 88(7):1108–1123, julho de 2000.
- Porche, I., Sampath, M., Sengputa, R., Chen, Y., e Lafortune, S. “A Decentralized Scheme for Real-time optimization of Traffic Signal”. In *Proceedings of the 1996 IEEE International Conference on Control Applications*, pp. 582–589, 1996.

- Puri, A. e Varaiya, P. “Verification of hybrid systems using abstractions”. In Antsaklis et al. [1995], pp. 359–369. ISBN 3-540-60472-3.
- Puri, A. e Varaiya, P. “Driving Safely in Smart Cars”. In *Proceedings of the American Control Conference*, pp. 3597–3599, 1995b.
- Raisch, J. e O’Young, S. D. “Discrete approximation and supervisory control of continuous systems”. *IEEE Transactions on Automatic Control, Special Issue on Hybrid Control Systems*, 43(4):569–573, abril de 1998.
- Robertson, D. I. “TRANSYT: Method for area traffic control”. *Traffic Engineering and Control*, 11: 276–281, 1969.
- Seibel, C. W., Farines, J.-M., e Cury, J. E. R. “Towards using hybrid automata for the mission planning for unmanned aerial vehicles”. In Antsaklis et al. [1999], pp. 324–340. ISBN 3-540-65643-X.
- Silva, B. I., Richeson, K., Krogh, B., e Chutinan, A. “Modeling and verifying hybrid dynamic systems using *CheckMate*”. In Engell, S., Kowalewski, S., e Zaytoon, J., editors, *Proceedings of the 4th International Conference on Automation of Mixed Processes: Hybrid Dynamic Systems (ADPM 2000)*, pp. 323–329, 2000.
- Silva, B. I., Stursberg, O., Krogh, B. H., e Engell, S. “An assessment of the current status of algorithmic approaches to the verification of hybrid systems”. In *Proceedings of 40th IEEE Conference on Decision and Control*, volume 3, pp. 2867–2874, Orlando, Florida, USA, dezembro de 2001.
- Stursberg, O. e Kowalewski, S. “Approximating switched continuous systems by rectangular automata”. In *Proceedings of European Control Conference, ECC’99*, Karlsruhe, Germany, setembro de 1999.
- Tavernini, L. “Differential Automata and their discrete simulators”. *Nonlinear Analysis, Theory, Methods and Applications*, 11(6):665–683, 1987.
- Tolba, C., Lefebvre, D., Thomas, P., e Moudni, A. E. “Continuous Petri nets models for analysis of traffic urban networks”. In *Proceedings of IEEE Conference on Systems, Man, and Cybernetics*, volume 2, pp. 1323–1328, 2001.
- Tomlin, C., Pappas, G., e Sastry, S. “Conflict resolution for Air Traffic Management : A study in multi-agent hybrid systems”. *IEEE Transactions on Automatic Control, Special Issue on Hybrid Control Systems*, 43(4):509–521, abril de 1998.
- Tomlin, C. e Greenstreet, M. R., editors. *Proceedings of Hybrid Systems: Computation and Control (HSCC 2002)*, volume 2289 de *Lecture Notes in Computer Science*, Stanford, CA, USA, março de 2002. Springer-Verlag. ISBN 3-540-43321-X.
- Tomlin, C. J., Mitchell, I., Bayen, A. M., e Oishi, M. “Computational techniques for the verification of Hybrid Systems”. *Proceedings of the IEEE*, 91(7):986–1001, julho de 2003.

- van der Schaft, A. e Schumacher, H. *An Introduction to Hybrid Dynamical Systems*. Number 251 in Lecture Notes in Control and Information Sciences (LNCIS). Springer-Verlag, London, England, 2000.
- van Egmond, R. J. e Olsder, G. J. “The (MAX,+) Algebra Applied to Synchronization of Traffic Light Processes”. In *Proceedings of WODES’98*, pp. 451–456, Cagliari, Italy, 1998.
- Varaiya, P. “Smart Cars on Smart Roads: problems of control”. *IEEE Transactions on Automatic Control*, 38(2):195–207, fevereiro de 1993.
- Villani, E. *Modelagem e análise de sistemas supervisórios híbridos*. Tese (doutorado), Escola Politécnica – Universidade de São Paulo, São Paulo, SP, 2003.
- Wimpey, D. G. *Finite-State control of discrete-time continuous processes: an automata motivated approach*. Tese (doutorado), Massachusetts Institute of Technology, Dept. of Electrical Engineering and Computer Science, Cambridge, junho de 1982.
- Witsenhausen, H. S. “A class of hybrid-state continuous-time dynamic systems”. *IEEE Transactions on Automatic Control*, 11(2):161–167, fevereiro de 1966.
- Wonham, W. M. *Linear Multivariable Control: A Geometric Approach*. Springer, 3 edition, 1985.
- Yovine, S. “Kronos: A Verification Tool for Real-Time Systems”. *International Journal of Software Tools for Technology Transfer*, 1:123–133, outubro de 1997. Springer-Verlag.